

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE TEORÍA DE LA SEÑAL Y COMUNICACIONES

INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN:
SISTEMAS DE TELECOMUNICACIÓN



PROYECTO FIN DE CARRERA

CLASIFICADOR DE ESTILOS DE NARRACIÓN BASADO EN APRENDIZAJE ESTADÍSTICO

Autor: Cristina González Ovejero

Tutor: José Miguel Leiva Murillo

Leganés, Septiembre de 2015

EL TRIBUNAL

Presidente:

Secretario:

Vocal:

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día, Facultad de la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la calificación de:

Fdo. Presidente

Fdo. Secretario

Fdo. Vocal

*“Si tienes vocación de águila, no te
contentes con ser ave de corral”*

AGRADECIMIENTOS

En primer lugar quería dar las gracias a mi familia, en especial a mi madre y a mis hermanas. Gracias a ellas soy lo que soy. Gracias por vuestro cariño y por vuestro apoyo constante. A mis cuñados por cuidarme como hermanos mayores, y a mis sobrinos por darme toda la alegría que a veces me falta, os quiero mucho a todos.

Gracias a ti, Jorge, porque sé que no ha sido fácil aguantarme en esta última etapa. Porque me has ayudado tanto material como anímicamente, porque sé que te alegras tanto o más que yo de que por fin termine esta etapa, GRACIAS, de verdad.

A mis amigos. A los de toda la vida: Mary, Huerty, Raquel, Rebe, Javi, Mikel, Miguel A... por estar siempre ahí, por tirar de mí cuando más lo he necesitado. A mis Charlys, porque hicieron que mi etapa universitaria fuera increíble. A mis compañeros de universidad, sobretodo los de los últimos años, Ana, Alex, Jorge... A Lourdes, por nuestras risas diarias.

Gracias Kira, porque tu sola presencia me tranquiliza, por nuestras largas salidas para desconectar, por tus miradas que no dicen nada y a la vez lo dicen todo.

Por último quería dar las gracias a José Miguel y a Emilio, porque sin vuestra ayuda no hubiera podido sacar este proyecto adelante, gracias por vuestra paciencia infinita y total disponibilidad.

RESUMEN

En la actualidad, el elevado volumen de datos disponible en Internet crea la necesidad de inventar sistemas automáticos que organicen dicha información. Es habitual que los programas de radio y televisión suban sus grabaciones a la red para que los espectadores puedan acceder a ellas en cualquier momento, por lo que se hace indispensable diseñar un sistema de indexado de audios que facilite la búsqueda a los usuarios. Por este motivo en este trabajo nos centraremos en la clasificación de audios de diferentes estilos de narración.

El presente proyecto propone diseñar un clasificador capaz de identificar de qué habla un locutor teniendo en cuenta únicamente el tono de voz y no el lenguaje que utiliza, es decir, que sea apto para diferenciar entre distintos estilos de narración. Para ello se implementarán distintos métodos de aprendizaje máquina haciendo uso del software matemático Matlab:

- **Máxima Verosimilitud** en el que se asumirá como modelo probabilístico una gaussiana multidimensional cuya media y covarianza serán calculadas a partir de los datos de entrenamiento.
- **Vecino más Próximo**, utilizando la técnica de validación cruzada se elegirá la K óptima, es decir, el número de vecinos que se tendrán en cuenta para decidir a que clase pertenece cada muestra.
- **Máquinas de Vectores de Soporte**, en el que también se hará uso de la técnica de validación cruzada para calcular los parámetros C y gamma óptimas de las funciones kernel que se utilizarán para encontrar los planos que separan las distintas clases.

A partir de un conjunto de audios de cuatro estilos de narración elegidos: Cuentos, Noticias, Fútbol y Monólogos, se comprobará la eficiencia de cada uno de los sistemas diseñados realizándose diferentes experimentos con cada uno de ellos. Comparando los resultados obtenidos se verificará que el clasificador que obtiene los mejores porcentajes a la hora de clasificar las cuatro clases es el de Máxima Verosimilitud.

Por último, se analizarán los resultados conseguidos con cada una de las clases pudiéndose ver cuáles consiguieron mayor porcentaje de aciertos y por el contrario, cuáles tuvieron más dificultades para ser diferenciadas del resto.

ABSTRACT

Nowadays, the large volume of data available on Internet creates the requirement to invent automatic systems that organize the information. Radio and television programs usually upload their recordings to the network so that viewers can access them at any time, so it is essential to design a system of audio indexing to do the search easier for users. Therefore in this paper we will focus on the classification of audios of different styles of narration.

This project aims at designing a classifier able to identify the topic of the speech only considering the tone of the speaker's voice and not the language used, i.e., if the system can distinguish between diverse styles of narration. With this in mind, I will implement different methods of machine learning using the mathematical software Matlab:

- **Maximum Likelihood**, where you assume a multivariate Gaussian as probabilistic model whose mean and covariance are calculated from the training data.
- **K-Nearest neighbour**, using cross-validation technique to selected the optimal K, i.e., the number of neighbours to be taken to decide to which class each sample belongs.
- **Support Vector Machines**, again using cross-validation technique for calculating the optimal C and gamma of the kernel functions that will be used to find the boundary that separates different classes.

From a set of audios of four selected styles of narrations: Stories, News, Soccer and Monologues, the efficiency of the designed systems will be checked by performing different experiments with them. Comparing the obtained results shows that the classifier that gets the best rates when classifying the four classes is the Maximum Likelihood.

Finally, the results obtained with each of the classes will be analysed to see which of them achieved best results and, on the other hand, which class had more difficulties to be distinguished from the rest.

ÍNDICE

| | |
|---|-----------|
| 1. INTRODUCCIÓN..... | 14 |
| 1.1 Motivación | 14 |
| 1.2 Objetivos | 15 |
| 1.3 Estructura de la memoria | 16 |
| 2. REVISIÓN DEL ESTADO DEL ARTE | 17 |
| 2.1 Introducción a la teoría de la decisión | 17 |
| 2.2 Aprendizaje automático (Machine Learning) | 18 |
| 2.2.1 Máquinas de Vectores de Soporte..... | 20 |
| 2.2.1.1 Caso linealmente separable | 21 |
| 2.2.1.2 Caso no linealmente separable | 24 |
| 2.2.1.3 SVM no lineal | 26 |
| 2.2.1.4 Clasificadores multiclase a partir de clasificadores binarios | 28 |
| 2.2.2 K-vecinos más próximos | 30 |
| 2.2.2.1 Algoritmo..... | 30 |
| 2.2.2.2 Elección del parámetro K | 32 |
| 2.2.3 Máxima Verosimilitud | 33 |
| 2.3 Extracción de características | 35 |
| 2.3.1 Mel Frequency Cepstral Coefficients | 35 |
| 2.3.2 Modelos AR | 38 |
| 3. DISEÑO E IMPLEMENTACIÓN DE LOS CLASIFICADORES | 39 |
| 3.1 Introducción | 39 |
| 3.2 Implementación: MATLAB | 39 |
| 3.3 Diseño | 40 |
| 3.4 Fragmento de audios | 40 |

| | | |
|-----------|--|-----------|
| 3.5 | Extracción de Características de Audio | 41 |
| 3.6 | Clasificadores | 42 |
| 3.6.1 | Máxima Verosimilitud (ML) | 42 |
| 3.6.2 | Vecino más próximo (KNN) | 44 |
| 3.6.3 | Máquinas de Vectores de Soporte (SVM) | 44 |
| 3.6.3.1 | <i>Librería LIBSVM</i> | 44 |
| 3.6.3.2 | <i>Diseño e implementación SVM</i> | 46 |
| 3.7 | Presentación de resultados | 48 |
| 4. | EXPERIMENTOS Y RESULTADOS | 49 |
| 4.1 | Introducción | 49 |
| 4.2 | Base de datos | 49 |
| 4.3 | Experimentos realizados | 50 |
| 4.4 | Resultados obtenidos | 52 |
| 4.4.1 | Resultados clasificador de Máxima Verosimilitud | 52 |
| 4.4.1.1 | <i>Experimento 1</i> | 53 |
| 4.4.1.2 | <i>Experimento 2</i> | 58 |
| 4.4.1.3 | <i>Experimento 3</i> | 60 |
| 4.4.1.4 | <i>Experimento 4</i> | 60 |
| 4.4.2 | Resultados clasificador Vecino más próximo | 74 |
| 4.4.2.1 | <i>K óptima</i> | 74 |
| 4.4.2.2 | <i>Experimento 1</i> | 75 |
| 4.4.2.3 | <i>Experimento 2</i> | 76 |
| 4.4.2.4 | <i>Experimento 3</i> | 77 |
| 4.4.2.5 | <i>Experimento 4</i> | 78 |
| 4.4.3 | Resultados clasificador Máquina de Vectores de Soporte | 81 |
| 4.4.3.1 | <i>C y gamma óptimas</i> | 81 |
| 4.4.3.2 | <i>Experimento 1</i> | 82 |
| 4.4.3.3 | <i>Experimento 2</i> | 85 |
| 4.4.3.4 | <i>Experimento 3</i> | 86 |
| 4.4.3.5 | <i>Experimento 4</i> | 88 |

| | | |
|---------------------|---|------------|
| 4.5 | Comparativa de los tres clasificadores..... | 95 |
| 4.5.1 | Experimento 1 | 95 |
| 4.5.2 | Experimento 2 | 97 |
| 4.5.3 | Experimento 3 | 98 |
| 4.5.4 | Experimento 4 | 98 |
| 5. | CONCLUSIONES Y LÍNEAS FUTURAS | 104 |
| 5.1 | Conclusiones | 104 |
| 5.2 | Líneas futuras | 107 |
| APÉNDICE | | 108 |
| | Clasificador ML | 108 |
| | Clasificador KNN | 121 |
| | Clasificador SVM | 125 |
| BIBLIOGRAFÍA | | 135 |

CAPÍTULO 1 INTRODUCCIÓN

En este primer capítulo se hablará de la motivación que nos ha llevado a realizar este trabajo, posteriormente se marcarán los objetivos que se pretenden conseguir y por último se explicará la forma en que está estructurada la memoria.

1.1 Motivación

En el mercado existen numerosos sistemas que son capaces de procesar la voz que emite el ser humano y reconocer las palabras que indican para poder transcribirlas o hacer algún tipo de acción: responder preguntas, activar o desactivar algún servicio, realizar recomendaciones... Otros métodos son capaces de identificar automáticamente a una persona a través de su voz. Estos últimos son empleados habitualmente en los controles de acceso a instalaciones o a un ordenador privado, en transacciones de autenticación, servicios personalizados, etc. En este proyecto se ha estudiado la posibilidad de crear un sistema que, sin atender a las palabras que se dicen, ni a quien las dice, sea capaz de distinguir de qué trata solo por el tono de voz.

En el presente, el volumen de datos disponible en Internet se hace cada vez más difícil de manejar, por lo que resulta interesante poder crear sistemas de clasificación automática para hacer más sencilla la búsqueda a los usuarios de lo que realmente les pueda interesar.

Los programas de radio y televisión, por ejemplo, frecuentemente suben sus grabaciones a la red en forma de podcasts o vídeos para poder tener acceso a ellas en cualquier momento. Se crea la necesidad, por tanto, de crear un método para indexar todos estos archivos de manera que se facilite su consulta. El procedimiento llevado a cabo en este proyecto podría ser de gran ayuda para poder organizar estos archivos por distintas temáticas.

Este proceso podría tener a su vez otras aplicaciones. Serviría de ayuda para discapacitados, por ejemplo, para preparar el material de personas invidentes o que se encuentren impedidas físicamente. Sería interesante como complemento a los métodos que ya se utilizan habitualmente de reconocimiento del habla, es decir que, aparte de interpretar el lenguaje, se analizara el tono para ayudar a la máquina a entender lo que estas personas necesitan, reduciendo el tiempo de respuesta.

Podría ser utilizado a su vez en el mundo de la publicidad, estudiando si de alguna manera se puede asociar el estilo de voz con el éxito del anuncio... una vez más no por las palabras que utilizan para vender un producto, sino por la forma de decirlas. Y así muchas más aplicaciones combinadas con otras ya creadas: en GPS, menús de aplicaciones, *audiobooks*, videojuegos, anotaciones de audio para subtitular películas, etc.

1.2 Objetivos

El objetivo principal de este proyecto es diseñar e implementar un sistema para la clasificación de diferentes estilos de narración. Se pretende comprobar si es posible clasificar fragmentos de audio teniendo en cuenta únicamente la onda sonora que genera el locutor independientemente del lenguaje que utiliza. Para ello no se van a emplear transcripciones de los audios a clasificar, solamente se va a extraer información sonora del audio.

A lo largo de este trabajo se evaluarán distintas técnicas de clasificación basadas en estadística y en aprendizaje automático que serán comparadas entre sí para poder afirmar con cuál de ellas se obtiene mejores resultados.

Otro de los objetivos es el de evaluar si hay estilos de narración que se mezclan más que otros, es decir, que tienen más problemas a la hora de diferenciarse del resto. También se quiere constatar si los clasificadores lo hacen mucho mejor en un género que en otros y si en general coinciden tanto en el estilo mejor clasificado como en el peor. Los estilos elegidos serán cualitativamente diferentes para poder hacer una discusión desde el punto de vista de la aplicación y no desde el del aprendizaje automático.

Al finalizar el proyecto se podrá ver si existe alguna técnica que consiga resultados aceptables para poder afirmar que es posible diseñar un sistema de clasificación que consiga lo mencionado anteriormente. Asimismo se confirmará cuál de las cuatro clases seleccionadas se clasifica mejor, y por el contrario, cuál se confunde más con el resto. Por otro lado también se verificará si los resultados son mejores en la clasificación por trozos o por audios y cuál es la mejor manera de combinar los datos de entrada al sistema para obtener los resultados más óptimos.

1.3 Estructura de la memoria

El proyecto se divide en cinco capítulos y un apéndice. En el Capítulo 1, tal y como hemos podido ver, se explican las motivaciones por las que se ha llevado a cabo este trabajo, los objetivos principales del proyecto así como la forma en la que se va a organizar la memoria.

En el Capítulo 2 se hará, en primer lugar, una introducción a la teoría de la decisión para después centrarse en el aprendizaje automático. Seguidamente se realizará un repaso de los tres clasificadores usados en este proyecto, así como las técnicas utilizadas para la extracción de características de audio.

El Capítulo 3 se centra en el diseño y la implementación de los clasificadores vistos en el Capítulo 2, así como en una descripción del software matemático utilizado.

En el Capítulo 4 se presenta, en primer lugar, la base de datos empleada, seguida de una breve descripción de los experimentos llevados a cabo. Por último se exponen los resultados obtenidos por los diferentes clasificadores en forma de matriz de confusión y se realizan comentarios al respecto.

El Capítulo 5 se realiza un resumen del trabajo realizado en este proyecto, se exponen las conclusiones finales y se mencionan posibles líneas futuras para continuar este trabajo.

Este proyecto también consta de un apéndice donde se muestran los resultados de los experimentos realizados adicionalmente que confirman las suposiciones estimadas a lo largo del presente estudio.

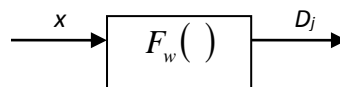
CAPÍTULO 2 REVISIÓN DEL ESTADO DEL ARTE

En este capítulo se realizará una breve introducción a la teoría de la decisión para después centrarse en el aprendizaje automático. Posteriormente se hará un repaso de los tres tipos de clasificadores seleccionados para este proyecto y las técnicas utilizadas para la extracción de características.

2.1 Introducción a la teoría de la decisión

Decidir es, dada una observación, encontrar a qué clase pertenece de entre un número finito y fijo de clases. Es decir, se dispone de varias clases, estas clases generan datos y lo que se quiere saber es, para un dato particular, cuál es la clase que lo generó. El problema de la decisión puede abordarse de dos maneras diferentes:

- 1) **Decisión automática o máquina (*Machine Learning*):** se utiliza cuando se dispone de un conjunto etiquetado de datos de entrenamiento (x) que proporciona información acerca de cuál sería la salida deseada del sistema para diferentes valores de las observaciones de entrada.



Según el resultado de $F_w()$ se decide, es decir, se toma la j -ésima decisión D_j . F_w es el clasificador o función clasificadora, siendo los w (w_1, w_2, \dots, w_N) los parámetros del clasificador. Según el valor que dé F_w para la x a la que se aplique decidiremos una cosa u otra.

Ejemplo:

Decidiremos D_1 para $F_w(x) > 0$ y D_0 para $F_w(x) < 0$

El objetivo del decisor construido es que sea capaz de proporcionar decisiones acertadas cuando sea aplicado a nuevos datos no vistos durante el entrenamiento. A esta propiedad se la conoce como capacidad de generalización del decisor.

Dentro del Machine Learning tenemos tres opciones:

- 1.1) Discriminativa paramétrica: en este caso se asume una función que clasifica, por ejemplo un plano (tal y como veremos en las Máquinas de Soporte Vectorial) y se aprenden los parámetros de esta función con la colección de datos.
 - 1.2) Discriminativa no paramétrica: no se requiere la definición a priori de ningún tipo concreto de función que implemente el clasificador, simplemente se asume un algoritmo de clasificación que usa los datos de los que se dispone, en este proyecto veremos el algoritmo del **vecino más próximo** (KNN).
 - 1.3) Generativa paramétrica: Se trata de intentar hacer algo similar a la decisión analítica (ver punto 2) pero en este caso no se conocen las densidades de probabilidad, por lo que se emplean los datos de los que se disponen para aprender las densidades (por eso se denomina generativo, porque se usan los datos para aprender la densidad de probabilidad que los generó). Lo que sí que se debe asumir es la forma de dicha densidad. Una vez que se han usado los datos para aprender (estimar) la densidad, se puede aproximar analíticamente. En este trabajo emplearemos el clasificador de **Máxima Verosimilitud**, también llamado ML por sus siglas en inglés (Maximum Likelihood).
- 2) **Decisión analítica (Statistics)**: la diferencia fundamental con el punto anterior consiste en que ahora si se conoce la “física” del problema, es decir, se dispone de información estadística que relaciona las observaciones y la información deseada, con lo cual se podrá establecer funciones de densidad de probabilidad que generan los datos y calcular analíticamente.

Este proyecto se centrará en la decisión automática, por lo que a continuación se realizará un repaso del aprendizaje automático para después centrarse en cada uno de los clasificadores elegidos.

2.2 Aprendizaje automático (Machine Learning)

El **aprendizaje automático** es una rama de la Inteligencia Artificial que abarca diferentes técnicas que permiten dotar a los computadores de la capacidad de "aprender" modelos computacionales tales que, de forma automática, les permitan resolver nuevos problemas o mejorar su comportamiento en problemas ya vistos. Es, por lo tanto, un proceso de inducción del conocimiento. En muchas ocasiones el campo de actuación del aprendizaje automático se solapa con el de la [estadística](#), ya que las dos disciplinas se basan en el análisis de datos.

¿Dónde y para qué se puede usar el aprendizaje automático?

- Tareas difíciles de programar (adquisición de conocimiento, reconocimiento de caras, voz, ...)
- Aplicaciones auto adaptables (interfaces inteligentes, spam filters, sistemas recomendadores, ...)
- Minería de datos/Descubrimiento de conocimiento (análisis de datos inteligente)

Se puede decir que existe una división primaria en el concepto de la clasificación automática:

- clasificación supervisada
- clasificación no supervisada.

La diferencia fundamental entre ambos métodos estriba en si se conoce o no la clase a la cual pertenece cada observación de los datos.

La **clasificación es supervisada** si ya existe un conjunto de observaciones clasificadas en un conjunto de clases dado, y se conoce la clase a la que cada observación pertenece.

En la clasificación supervisada se distinguen dos fases fundamentales bien diferenciadas:

1. Desarrollo o creación de una o varias reglas de decisión (diseño del clasificador)
2. El proceso en sí de clasificación de nuevas observaciones.

En la primera fase, el conjunto cuyas clases ya están bien definidas se desglosa en un conjunto de entrenamiento y otro de test. Se diseña el clasificador con el conjunto de entrenamiento y se observa su capacidad para clasificar con el conjunto de test. En la segunda fase se procede a clasificar nuevas observaciones de las que se desconoce la clase a la que pertenecen.

La **clasificación es no supervisada** cuando se dispone de un conjunto de observaciones donde se desconoce tanto el número de clases en que es razonable clasificarlo así como a qué clase pertenece cada observación.

Este proceso de clasificación no supervisada, es significativamente más complejo que el de la supervisada ya que se desconocen las clases naturales, y dependerá de la habilidad para seleccionar las características que representan al objeto (elección de las variables que constituyen una observación) y la metodología de clasificación.

2.2.1 Máquinas de Vectores de Soporte

Las **máquinas de soporte vectorial** o **máquinas de vectores de soporte** (SVM por su nombre en inglés *Support Vector Machines*) son un conjunto de algoritmos de clasificación supervisada desarrollados por V.Vapnik y A.Lerner en los laboratorios AT&T en 1963, aunque no fue hasta los años 90 cuando fueron presentados en la conferencia Colt (Boser, Guyon y Vapnik, 1992). Posteriormente, en 1995, Cortés y Vapnik introdujeron el clasificador de margen blando y ese mismo año fue extendido por Vapnik al caso de regresión.

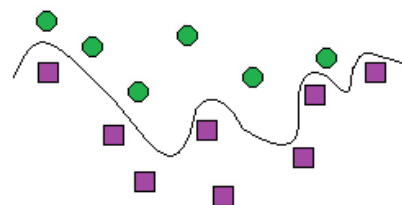
Las SVM han mostrado tener gran desempeño en muchas aplicaciones, más que las máquinas de aprendizaje tradicional como las redes neuronales. Debido a sus prestaciones y a su versatilidad han sido introducidas como herramientas poderosas para resolver problemas de clasificación.

Fueron originalmente ideadas para la resolución de problemas de clasificación binaria en los que las clases eran linealmente separables. El primer paso, antes de poder clasificar, es realizar una etapa de entrenamiento. Consiste en encontrar el hiperplano $h(x) = 0$ que separe de forma óptima un conjunto de datos $X \in \mathbb{R}^d$ según la clase $Y \in \{-1,1\}$ a la que pertenecen. Se seleccionará el hiperplano que mejor represente el límite entre los dos conjuntos, es decir, el que maximiza el margen m entre las muestras de cada clase respecto a la frontera de separación.

El objetivo es que tras este entrenamiento, la máquina generalice bien para datos nuevos (aquellos que no han participado en el entrenamiento) y sea capaz de indicar a que clase pertenece cada una de las muestras. Para realizar una buena clasificación se busca seleccionar una función f que por un lado minimice el coste empírico sobre el conjunto de entrenamiento y por otro mantenga una correcta generalización de la máquina. Es decir, a partir de unos datos de entrada x_i , las SVM nos proporcionarán su clase según la función de clasificación $f(x_i) = \text{signo}(h(x_i))$.

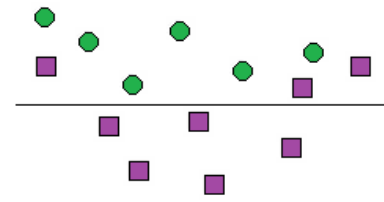
Para obtener una buena generalización, hay que tratar de evitar dos de los efectos típicos del entrenamiento:

- ✓ **Sobreajuste:** número de parámetros excesivamente alto, que lleva a una solución que incluye demasiado detalle particular del conjunto de entrenamiento, por lo que se crean problemas a la hora de generalizar y al introducir un dato nuevo no lo clasifica de la forma correcta.



Ejemplo de sobreajuste

- ✓ **Subajuste:** insuficiencia en el número de parámetros, el hiperplano trazado es demasiado sencillo y no logra una buena generalización, por lo que la máquina tendrá problemas a la hora de clasificar nuevas muestras.



Ejemplo de subajuste

A continuación se realizará una revisión de la teoría básica de las SVM en problemas de clasificación.

2.2.1.1 Caso linealmente separable

Se considera que un problema es separable linealmente cuando para cualquier conjunto de muestras existe un hiperplano único capaz de clasificarlas con error cero. Por lo tanto, el objetivo principal en este caso es el de encontrar un plano de separación que es definido con los vectores soporte, siendo estos últimos muestras críticas que salen del problema de optimización.

Posteriormente se introducen multiplicadores de Lagrange para resolver la optimización que encuentra el clasificador. La mayoría de los coeficientes de Lagrange serán nulos y los únicos coeficientes que serán distintos de cero serán aquellos puntos que estarán situados exactamente a la distancia señalada por el margen, estos son, como se indicó anteriormente, los Vectores Soporte, los cuales actuarán como representantes de todo el conjunto de datos (la solución va a depender únicamente de ellos). Si entrenamos la máquina solamente teniendo en cuenta estos vectores, se van a obtener los mismos resultados que si utilizamos el conjunto total de los datos.

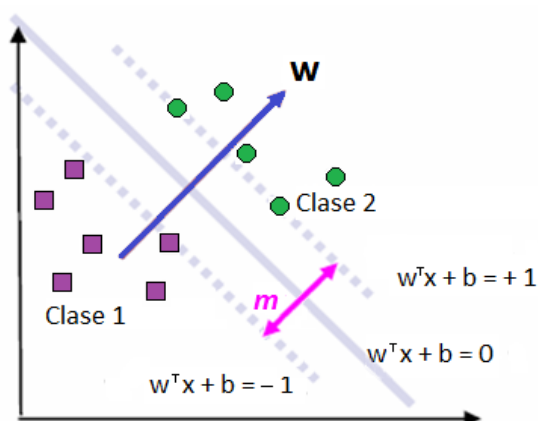


Figura 1. Separación de datos mediante SVM. La frontera de decisión debe estar tan lejos de los datos de ambas clases como sea.

En la Figura 1 se observa un caso binario separable linealmente. Se representan datos de dos clases (cuadrados morados y círculos verdes) separados por el hiperplano que maximiza la distancia entre ellos. Esta distancia es la marcada como margen m , que es máxima para el hiperplano obtenido en este caso, cualquier otro hiperplano presentaría un margen de separación de clases menor y por lo tanto sería menos adecuado.

Se tiene un conjunto de N datos, x_i , separables linealmente, con sus correspondientes etiquetas asociadas, y_i , que definirán a qué clase pertenece cada dato $\{(x_1 \cdot y_1), (x_2 \cdot y_2), \dots, (x_N \cdot y_N)\}$ donde $x_i \in \mathbb{R}^d$ e $y_i \in \{-1, 1\}$, siendo d el número de dimensiones o componentes de los vectores que contienen los datos.

Un clasificador es lineal si su función de decisión puede expresarse mediante una función lineal en x . Así pues, se puede usar como frontera de decisión un hiperplano $h(x_i)$ tal que:

$$h(x) = \omega^T x + b = 0 \text{ donde } \omega \text{ y } x \in \mathbb{R}^d \quad (1)$$

En (1) b es una constante que indica la posición del plano respecto al origen de coordenadas, que recibe el nombre de *sesgo*. ω es el vector normal al hiperplano y tiene la forma:

$$\omega = \sum_{i=1}^N \alpha_i y_i x_i \quad (2)$$

donde los α_i son los **multiplicadores de Lagrange**, los cuales serán nulos en su mayoría (como ya se ha comentado anteriormente) y solo se salvarán los que sean Vectores Soporte.

La clasificación se basa en determinar en qué zona del hiperplano se encuentra el punto a clasificar. Así pues, el clasificador puede representarse mediante las expresiones:

$$\begin{aligned} \omega^T x_i + b &> -1 \text{ para } y_i = +1, i=1, \dots, n \\ \omega^T x_i + b &< +1 \text{ para } y_i = -1, i=1, \dots, n \end{aligned} \quad (3)$$

Si el resultado de la operación es negativo la muestra pertenecerá a una clase, y si es positivo a la otra. Las expresiones mostradas anteriormente las podemos reducir a una sola:

$$y_i \{\omega^T \cdot x_i + b\} \geq 1 \quad (4)$$

Al ser un problema linealmente separable, el número de posibles hiperplanos que sean capaces de separar las muestras de ambas clases, son infinitos. En la Figura 2 se pueden ver 5 de las infinitas posibles soluciones. El que buscamos es aquel que tenga un mayor margen. Es decir, queremos maximizar la distancia entre los datos y la frontera de decisión.

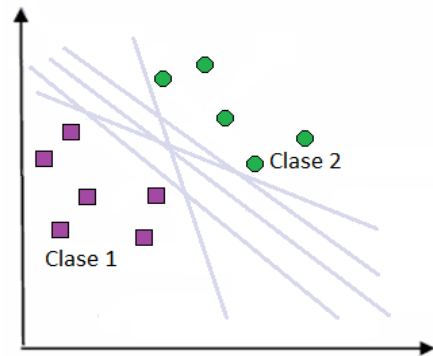


Figura 2. 5 posibles soluciones para el caso linealmente separable

Para conseguirlo, se considera que los vectores soporte (los puntos más cercanos al hiperplano), cumplen:

$$h(x_i) = x_i \cdot \omega + b = +1 \quad (5)$$

$$h(x_i) = x_i \cdot \omega + b = -1 \quad (6)$$

Estos dos hiperplanos son paralelos entre si y paralelos a su vez con la frontera de decisión. El margen será la distancia entre ambos. Lo que se busca aquí es potenciar la generalización, es decir, que una vez se haya entrenado a la máquina, esta sea capaz de clasificar correctamente los nuevos datos introducidos. Por lo tanto, es mejor cuanto mayor sea la distancia entre los datos y la frontera de clasificación.

La distancia $dist(h, x)$ de un punto al hiperplano es:

$$dist(h, x) = \frac{|h(x)|}{\|\omega\|} \quad (7)$$

Ya que los puntos más próximos al hiperplano cumplen $|h(x)| = 1$, su distancia al hiperplano sería:

$$dist(h, x) = \frac{1}{\|\omega\|} \quad (8)$$

Para encontrar los valores de ω y b hay que resolver un problema de optimización que consiste en maximizar $dist(h, x)$ entre el hiperplano y el punto de entrenamiento más próximo. Es decir, maximizar $1/\|\omega\|$ sujeto a $y_i \{ \omega^T \cdot x_i + b \} \geq 1, i=1, \dots, n$, la cuál es la condición de que ningún vector de entrenamiento quede dentro del margen que separa las dos clases.

Así pues, el margen del hiperplano $h(x_i)$ (distancia a los vectores más cercanos pertenecientes a diferentes clases) a maximizar es $2/\|\omega\|$, es decir, la suma de ambas distancias. Maximizar este margen es equivalente a minimizar la norma de ω . De esta forma, el problema de encontrar el hiperplano óptimo puede formularse como:

$$L(w) = \min \left\{ \frac{1}{2} \|\omega\|^2 \right\} \quad (9)$$

Empleando los multiplicadores de Lagrange se permite representar el hiperplano como combinación lineal de las propias muestras. Incorporando las restricciones (3) y (4), la expresión a minimizar queda así:

$$L_d = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i (y_i (\omega^T \cdot x + b) - 1) \quad (10)$$

Tras derivar dicha expresión con respecto a w y b e igualar a 0 para minimizarla, nos permite deducir dos ecuaciones:

$$w = \sum_{i=1}^N \alpha_i y_i x_i \quad (11)$$

$$\sum_{i=1}^N \alpha_i y_i = 0 \quad (12)$$

Teniendo en cuenta las ecuaciones anteriores, la máquina de vectores soporte, actuando como clasificador y para el caso en el que el problema sea linealmente separable, puede escribirse como:

$$f(x) = \text{sign}(\sum_{i=1}^N \alpha_i y_i (x_i x) + b) \quad (13)$$

Recordando lo que se dijo anteriormente, los Vectores de Soporte son aquellos datos que se encuentran justo a la distancia del margen y son los únicos que se necesitan para definir la frontera de decisión puesto que sus multiplicadores de Lagrange, a diferencia del resto, serán distintos de 0.

2.2.1.2 Caso no linealmente separable

Es posible que los datos de entrada no sean linealmente separables, es decir, que no sea posible trazar una frontera lineal para delimitar ambas clases de datos. Para este caso hay dos posibles soluciones: tratar de encontrar el hiperplano que cometa el menor número de errores o bien buscar una frontera no lineal.

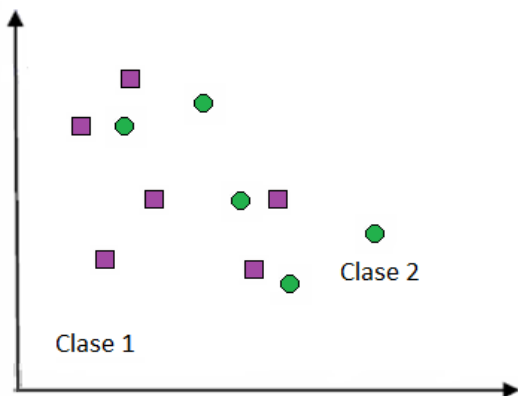


Figura 3. Caso no linealmente separable

Para el primero de los casos introduciremos unas variables positivas, ξ_i , en el problema de optimización, que son unas variables de holgura o también llamadas *slack*, las cuáles van a controlar el error permitido y a penalizar las muestras mal clasificadas. Así pues la ecuación (4) que utilizábamos para el caso separable linealmente va a ser modificada de la siguiente manera:

$$y_i \{\omega^T \cdot x_i + b\} \geq 1 - \xi_i \quad 1 \leq i \leq n \quad (14)$$

$$\xi_i \geq 0, \forall i \quad 1 \leq i \leq n \quad (15)$$

De esta forma, en los datos bien clasificados se cumplirá que $0 < \xi_i < 1$, dependiendo de cómo de cerca de la frontera esté la muestra, mientras que en los datos que se hayan clasificado mal $\xi_i > 1$. Con estas condiciones podremos colocar los datos outliers (valores atípicos) dentro del margen y así poder calcular el plano. Si no se hiciera uso de las variables *slack*, el problema de optimización no tendría solución, por lo que no habría plano.

En este caso la ecuación a minimizar (9) es modificada del siguiente modo:

$$L_p = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^N \xi_i \quad (16)$$

La cuál está sujeta a las condiciones (14) y (15). Siendo C una constante, que puede ser definida como un parámetro de regularización. Este es el único parámetro libre que debe ser ajustado en la formulación de la SVM. El ajuste de este parámetro puede hacer un balance en la maximización del margen y la minimización de los errores. En el presente proyecto, el valor de esta constante se ha calculado mediante **validación cruzada**, tal y como se explicará en el Capítulo 4.

Utilizando de nuevo multiplicadores de Lagrange, la ecuación (16) quedaría así:

$$L_{pd} = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \mu_i \xi_i - \sum_{i=1}^N \alpha_i (y_i (x_i \cdot \omega + b) - 1 + \xi_i) \quad (17)$$

Donde μ_i es el multiplicador de Lagrange que se introduce para forzar que ξ_i sea positiva. Si ahora, al igual que hicimos en el caso linealmente separable, derivamos la ecuación con respecto a cada una de las variables e igualamos a cero, obtendremos la solución óptima y:

$$C - \mu_i - \alpha_i = 0 \quad (18)$$

Por lo que el problema cuadrático queda de la forma:

$$L_{pd} = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \mu_i \xi_i - \sum_{i=1}^N \alpha_i (y_i (x_i \cdot \omega + b) - 1 + \xi_i) \quad (19)$$

2.2.1.3 SVM no lineal

En algunas ocasiones, la solución lineal de los dos apartados anteriores no obtiene un buen resultado. Para estos casos se va a necesitar trazar fronteras de clasificación no lineales, por lo que se recurrirá a proyectar los datos a un espacio de mayor dimensión donde la solución lineal sea válida, operación facilitada por la SVM no lineal.

La SVM no lineal se puede interpretar como una generalización del hiperplano de decisión. Para el caso en el que los datos no sean linealmente separables, se puede aplicar una transformación $\phi(x)$ sobre el espacio de trabajo para obtener un **espacio de características** de mayor dimensión donde si se podrá encontrar un hiperplano que los separe. La formulación es básicamente la misma, únicamente se reemplazará x por $\phi(x)$ (el producto escalar se hará en el espacio de características en lugar de en el espacio de entrada).

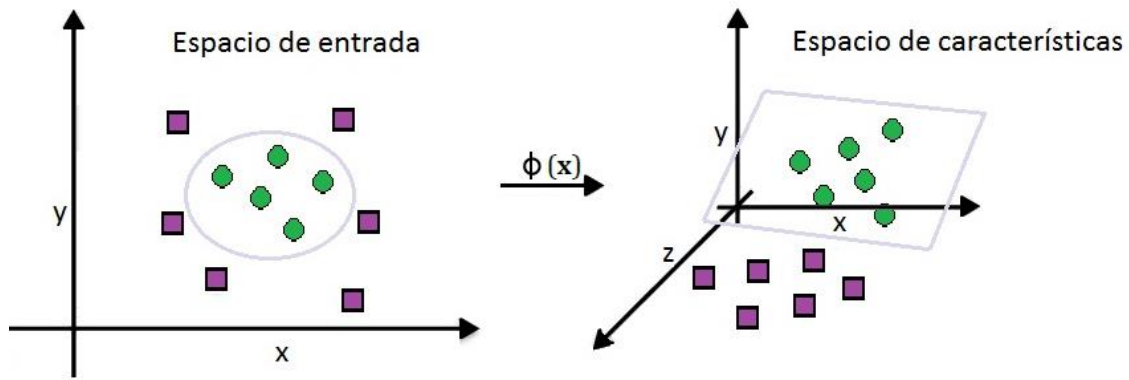


Figura 4: Transformación espacial del espacio de entrada

Para construir una SVM en el espacio resultante, es importante tener en cuenta que este debe ser un *Espacio de Hilbert*, donde la proyección del espacio de entrada se lleva a cabo mediante una Función núcleo o Función Kernel que debe cumplir:

$$k(x, x_i) = \langle \phi(x), \phi(x_i) \rangle \quad (20)$$

Esta función permite aplicar el algoritmo de entrenamiento de las SVM sin conocer la transformación ϕ .

Para saber si una función $k(x, x_i)$ es válida para poder aplicar el procedimiento hay que ver si cumple el *Teorema de Mercer*, el cuál expone que para garantizar que la función sea un producto interno en un espacio de alta dimensionalidad, se debe cumplir la condición:

$$\iint k(u, v) g(u) g(v) du dv > 0 \quad (21)$$

Es decir, la función kernel debe ser una función definida positiva.

Con el fin de obtener los mejores resultados, existen distintas funciones kernel que permiten adaptar la SVM a cada conjunto de datos. Las usadas más habitualmente son:

- Lineal:

$$k(x, x_i) = x \cdot x_i \quad (22)$$

- Polinómica:

$$k(x, x_i) = (\gamma x \cdot x_i + c)^\alpha \quad (23)$$

- Gaussiana:

$$k(x, x_i) = \exp(-\gamma |x - x_i|^2) \quad (24)$$

Siendo c un coeficiente, γ una constante de proporcionalidad y α el rango del polinomio. En este proyecto se va a realizar una comparativa entre la kernel lineal y la kernel gaussiana.

Usando la SVM no lineal el coste en el que se incurre debido a la dimensionalidad es muy bajo, esto es debido a que no es necesario trabajar en el espacio de características, basta con conocer su producto escalar.

Tras la transformación realizada y las consideraciones anteriores, la función obtenida en el caso linealmente separable (13) se redefine de la siguiente manera:

$$f(x) = \text{sign}(\sum_{i=1}^N \alpha_i y_i \cdot k(x, x_i) + b) \quad (25)$$

Teniendo en cuenta el Teorema de Mercer comentado anteriormente (21), el problema cuadrático de entrenar la SVM quedaría de la siguiente manera:

$$L_d = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad (26)$$

Que estará sujeta a las siguientes restricciones:

$$0 \leq \alpha_i \leq C \quad (27)$$

$$\sum_i \alpha_i y_i = 0 \quad (28)$$

El vector hiperplano resultante es:

$$w = \sum_{i=1}^N \alpha_i y_i \phi(x_i) \quad (29)$$

Puede suceder que el problema no sea separable tampoco en el nuevo espacio. Si esto sucediera, se actuaría de igual manera que en el problema lineal: se introduce un término de coste C y las pérdidas ξ_i . Así pues, reescribiendo las ecuaciones (14) y (15) para el caso no lineal, se obtiene:

$$y_i \{ \phi(x_i) \cdot w + b \} \geq 1 - \xi_i \quad (30)$$

$$\xi_i \geq 0, \forall i \quad (31)$$

De esta forma, la nueva función será la misma que en el caso lineal para problemas no separables:

$$\min_{w,b,\xi_i} \frac{1}{2} \|w\|^2 - C \sum_{i=1}^N \xi_i \quad (32)$$

Usando multiplicadores de Lagrange e introduciendo las ecuaciones (30) y (31) en (32), queda:

$$L_{pd} = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \mu \xi_i - \sum_{i=1}^N \alpha_i (y_i (\phi(x_i) \cdot w + b) - 1 + \xi_i) \quad (33)$$

Siendo una ecuación parecida a la obtenida en el caso no separable linealmente (17), en la que únicamente se ha sustituido x_i por $\phi(x_i)$.

2.2.1.4 Clasificadores multiclase a partir de clasificadores binarios

Es bastante habitual cuando se trabaja con problemas de clasificación extraídos de la vida real encontrarse con situaciones de multclasificación. Por otro lado, es bien sabido que la SVM es un clasificador binario por definición, es decir, solo puede resolver problemas en los que hay dos clases posibles, la +1 y la -1. Para solventar este problema se puede hacer uso de dos algoritmos que, a partir de clasificadores binarios, permiten encontrar una solución para el problema de clasificación de K clases:

1) Uno-contra-todos, en inglés one-against-rest (OAR):

- Es también conocido como el método estándar
- Se entrenan K clasificadores binarios (una clase es la positiva y el resto la negativa)
- Se predice la clase para todos los clasificadores
- La clase asignada es aquella con la que se consiguió mayor margen (en el caso en que se clasifique como positiva en más de un clasificador)

Aplicación de este algoritmo a un clasificador de 4 clases: Se dispone de 4 clases, por lo que se deben entrenar 4 SVMs que serán SVM1, SVM2, SVM3 y SVM4. La SVM_i se entrena haciendo que las muestras de la clase i sean la clase +1 de la SVM y las muestras de las otras 3 clases sean la clase -1.

Para clasificar un dato de test, se pasa por las 4 SVMs y se asigna a la clase cuya SVM_i dé la salida blanda más alta (la salida blanda es la que indica cómo de cerca o de lejos del margen está cada dato). Si el dato a clasificar está muy lejos del margen, esa salida será muy positiva y eso quiere decir que la SVM_i está muy segura de que ese dato es de la clase i. Si todas las SVMs dan una salida negativa es porque ninguna cree que ese dato sea de su clase. En ese caso la salida más alta (menos negativa) da la SVM que esté menos segura de que el dato no sea de su clase. Por tanto asignar un dato a la SVM con la salida más grande funciona en ambos casos.

2) Uno-contra-uno, en inglés one-against-one (OAO):

- Se construyen $k(k-1)/2$ clasificadores cada uno entrena datos de dos clases diferentes
- Se usa la estrategia de votación para clasificar: cada clasificador binario se considera como un voto y se toma la clase con mayor número de votos.

Aplicación de este algoritmo a un clasificador de 4 clases: Al disponer de 4 clases se tienen que entrenar 6 SVMs: SVM₁₂, SVM₁₃, SVM₁₄, SVM₂₃, SVM₂₄, SVM₃₄. La SVM_{ij} se entrena usando como muestras positivas las de la clase i y como muestras negativas las de la clase j.

Para clasificar un dato de test, se pasa por las 6 SVMs. Cada SVM vota por una clase, es decir asigna el dato a una de las dos clases con las que ha sido entrenada. Observando el nombre de cada SVM, se puede ver como cada clase recibirá 3 votos como máximo, por ejemplo la clase 4 puede ser votada por SVM₁₄, SVM₂₄ y SVM₃₄. El dato de test se asignará a la clase que tenga más votos. Si hay un empate entre dos clases i y j se puede desempatar mirando al clasificador SVM_{ij} o sumando las salidas blandas de cada clase. La probabilidad de que haya empates con 4 clases es bastante escasa.

2.2.2 K-vecinos más próximos

El de Vecino más próximo, en inglés *Nearest Neighbours* (K-NN) es uno de los algoritmos más simples y atractivos en reconocimiento de patrones. Fue desarrollado por Fix y Hodges (1951). Es un sistema de clasificación supervisado basado en criterios de vecindad. Recordemos que los sistemas de **clasificación supervisados** son aquellos en los que, a partir de un conjunto de ejemplos clasificados (**conjunto de entrenamiento**), se intenta asignar una clasificación a un segundo conjunto de ejemplos. En particular, k-NN se basa en la idea de que los nuevos ejemplos serán clasificados a la clase a la cual pertenezca la mayor cantidad de vecinos más cercanos del conjunto de entrenamiento más cercano a él.

El **algoritmo del vecino más cercano** para determinar a qué clase pertenecía la nueva muestra tenía en cuenta únicamente el vecino más próximo a ella, esto hacía pensar que no se estaba aprovechando de forma eficiente toda la información que podría ser extraída del conjunto de entrenamiento. Con el objetivo de resolver esta posible deficiencia surge la regla de los k vecinos más cercanos (k-NN), que es una extensión en la que se utiliza la información suministrada por los k ejemplos del conjunto de entrenamiento más cercanos.

Dependiendo del tipo de datos que queramos evaluar a veces se deben utilizar más o menos vecinos. Normalmente incluir más vecinos suele reducir el ruido de la clasificación pero también provoca que haya clases demasiado parecidas, siendo esto un problema a la hora de decidir la clase a la que pertenece la muestra evaluada. Para solucionar este inconveniente existen reglas que determinan qué hacer en caso de que dos clases empaten respecto a la pertenencia de una muestra. No obstante, la mejor manera de elegir el número de vecinos, es mediante el entrenamiento de las muestras.

2.2.2.1 Algoritmo

Las muestras de entrenamiento son vectores en un espacio de características multidimensional, cada uno con una etiqueta de clase. La fase de entrenamiento del algoritmo consiste solo en almacenar los vectores de características de las muestras de entrenamiento con sus respectivas etiquetas. Posteriormente se pasa a la fase de clasificación, que evalúa las muestras de las que se desconoce la clase según el patrón de los datos evaluados inicialmente, calculando las distancias y seleccionando los k vecinos más cercanos, que permitirá definir la clase a la que pertenecerá el nuevo elemento. K es una constante definida por el usuario, y cada vector de test es clasificado asignando la etiqueta que es más frecuente entre las K muestras de entrenamiento más cercanos al vector de test. La búsqueda de los K vecinos más cercanos de forma exhaustiva es costosa, pero no mucho más que la búsqueda de un único vecino más cercano, siempre que K sea menor que el valor del conjunto de entrenamiento.

El algoritmo más sencillo para la búsqueda del vecino más cercano es el conocido como fuerza bruta, que calcula todas las distancias de las muestras del entrenamiento respecto a una muestra concreta, y asigna como conjunto de vecinos más cercanos aquél cuya distancia sea mínima.

Los ejemplos de entrenamiento están descritos en términos de p atributos considerando q clases para la clasificación. Los valores de los atributos del i -ésimo ejemplo (donde $1 \leq i \leq n$) se representan por el vector p -dimensional

$$x_i = (x_{1i}, x_{2i}, \dots, x_{pi}) \in X \quad (34)$$

Para calcular la distancia entre los vecinos se suele utilizar la distancia euclidiana o euclídea:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^p (x_{ri} - x_{rj})^2} \quad (35)$$

Los requisitos que deben cumplir las funciones de distancias son:

1. $d(x_i, x_j) \geq 0$
2. $d(x_i, x_j) = 0$
3. $d(x_i, x_j) = d(x_j, x_i)$
4. $d(x_i, x_j) \leq d(x_i, x_h) + d(x_h, x_j)$

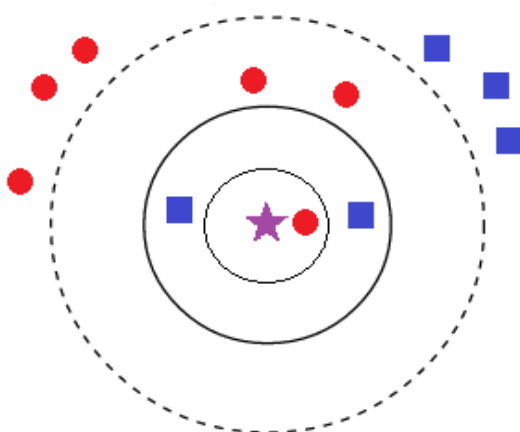
Un inconveniente del básico "voto mayoritario" para clasificación es que las clases más frecuentes tienden a dominar la predicción del nuevo vector de test, ya que tienden a aparecer en los K vecinos más cercanos debido a su mayor presencia en el conjunto de datos. Una manera de solucionar este problema es ponderar la clasificación teniendo en cuenta la distancia desde el punto de test a cada uno de sus K vecinos más cercanos.

2.2.2.2 Elección del parámetro K

De los datos de entrenamiento depende la mejor elección de k . Por norma general, valores grandes de k reducen el efecto de ruido en la clasificación, pero por otro lado crean fronteras entre clases parecidas. Un buen valor de K puede seleccionarse mediante una optimización de uso. En este proyecto se calculará K mediante una validación cruzada, esta es una técnica muy precisa porque se evalúa a partir de distintas combinaciones de datos de entrenamiento y test, pero tiene una desventaja y es que es un método demasiado lento desde el punto de vista computacional

En la validación cruzada de K iteraciones o K -fold cross-validation los datos de muestra se dividen en K subconjuntos. Uno de los subconjuntos se utiliza como datos de prueba y el resto ($K-1$) como datos de entrenamiento. El proceso de validación cruzada es repetido durante k iteraciones, con cada uno de los posibles subconjuntos de datos de prueba. Finalmente se realiza la media aritmética de los resultados de cada iteración para obtener un único resultado. Tal y como se explicará en el Capítulo 4 (apartado 4.3.2) se realizará una validación cruzada de 5 iteraciones.

En problemas prácticos donde se aplica esta regla de clasificación se acostumbra a tomar un número k de vecinos impar para evitar posibles empates (aunque esta decisión solo resuelve el problema en clasificadores binarios). En otras ocasiones, en caso de empate, se selecciona la clase que verifique que sus representantes tengan la menor distancia media al ejemplo que se está clasificando. En última instancia, si se produce un empate, siempre se puede decidir aleatoriamente entre las clases con mayor representación.



La Figura 5 muestra un ejemplo en el que se está intentando clasificar la estrella morada. Si con el clasificador k -NN se utiliza 1 vecino más cercano, la muestra pertenecerá a la clase formada por un círculo rojo. Sin embargo, si se utiliza el clasificador con los 3 vecinos más cercanos, el elemento evaluado pertenecerá a la clase formada por cuadrados azules, ya que 2 de los elementos que están más cerca de la muestra evaluada son cuadrados y una al

círculo. Si utilizáramos el clasificador con los 5 vecinos más cercanos (5-NN), el elemento evaluado será un círculo rojo, porque 3 elementos son círculos y 2 cuadrados.

2.2.3 Máxima Verosimilitud

Los métodos analíticos se basan en la disponibilidad de cierta información estadística que relaciona observaciones y valor a estimar. El tipo de información requerida para el diseño del estimador varía en función de cuál sea el tipo de estimador que se desea construir (por ejemplo, según sea el criterio de diseño). En general, esta aproximación analítica resulta posible cuando la naturaleza del problema permite determinar un modelo probabilístico de las variables involucradas.

La información necesaria para el diseño analítico será la siguiente:

- Probabilidades a priori $\rightarrow p_H(H_i) = P(H_i) \quad \forall i \rightarrow$ es la probabilidad (a priori) de que se dé la hipótesis H_i .
- Verosimilitudes $\rightarrow p_{x|H_i}(x) = p(x|H_i) \quad \forall i \rightarrow$ función de probabilidad que nos informa de la distribución de x cuando la hipótesis es H_i .
- Política de costes \rightarrow coste de tomar la decisión D_j cuando la hipótesis es $H_i \rightarrow C_{ji}(C(D_j, H_i))$

Otras relaciones importantes de estas variables a tener en cuenta serían:

- Densidad de probabilidad conjunta: $p(x, H_i) = p(x|H_i)p(H_i)$ (Habrà una para cada hipótesis)
- Densidad de probabilidad marginal: $p(x) = \sum_{i=1}^C p(x|H_i)p(H_i) = \sum_{i=1}^C p(x, H_i)$

Clasificador de Máximo a Posteriori (MAP):

Como su propio nombre indica, el clasificador de máximo a posteriori (*Maximum a Posteriori*, MAP) se define con el valor de H que maximiza la distribución de probabilidad a posteriori de dicha variable, es decir, el valor de H que concentra mayor densidad de probabilidad para cada valor de la variable observable:

$$D_j^*: j^* = \arg \max_j p(H_j | x) \quad (36)$$

Aplicando el teorema de Bayes a la distribución de probabilidad:

$$D_j^* : j^* = \arg \max_j \frac{p(\underline{x} | H_j) p(H_j)}{p(\underline{x})} \quad (37)$$

Lo que interesa principalmente es la probabilidad de error. Como el conjunto de entrenamiento que se utilizará en este proyecto es equilibrado, es decir, todas las clases están igual de pobladas, se puede decir que las probabilidades a priori son equiprobables, por lo que el MAP que da la mejor probabilidad de error es el equivalente al ML.

$$D_j^* : j^* = \arg \max_j p(\underline{x} | H_j) \quad (37)$$

Uno de los modelos probabilísticos más habituales es la Gaussiana multidimensional, que es definida de la siguiente manera:

$$p(\underline{x} | H_j) = G_x(\underline{m}, \underline{V}) = \left[(2\pi)^N |\underline{V}| \right]^{-1/2} \cdot e^{-\frac{1}{2}(\underline{x} - \underline{m})^T \underline{V}^{-1} (\underline{x} - \underline{m})}$$

Como se puede observar, la gaussiana queda completamente caracterizada por su media (\underline{m}) y su matriz de covarianzas (\underline{V}), siendo N el número de elementos y \underline{x} los datos de entrenamiento.

$$\underline{m}_{(Nx1)} = \begin{pmatrix} m_{x1} \\ m_{x2} \\ \dots \\ m_{xN} \end{pmatrix} \quad \underline{x}_{(Nx1)} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix} \quad \underline{V}_{(NxN)} = \begin{pmatrix} V_{x1} & V_{x1x2} & V_{x1x3} & \dots & V_{x1xN} \\ V_{x2x1} & V_{x2} & V_{x2x3} & \dots & V_{x2xN} \\ V_{x3x1} & V_{x3x2} & V_{x3} & \dots & V_{x3xN} \\ \dots & \dots & \dots & \dots & \dots \\ V_{xNx1} & V_{xNx2} & V_{xNx3} & \dots & V_{xN} \end{pmatrix}$$

2.3 Extracción de características

2.3.1 Mel Frequency Cepstral Coefficients

Este algoritmo fue originalmente propuesto por Davis y Mermelstein (1980) en un sistema de reconocimiento automático de voz. Actualmente es uno de los más utilizados en esta área. Ha demostrado también buenos resultados en extracción de información en música polifónica, aplicándose a la detección de instrumentos, de cantantes y de generación de listas de reproducción entre otros.

Los MFCC han sido las características dominantes utilizadas para reconocimiento de voz durante mucho tiempo. Su éxito ha sido debido a su capacidad para representar el espectro de la amplitud del habla de una forma compacta. Cada paso en el proceso de creación de características MFCC está motivado por consideraciones perceptivas o computacionales. En los siguientes párrafos se examinan estos pasos con más detalle. En [1] se puede consultar una descripción más completa de los supuestos del proceso.

La Figura 6 muestra el proceso de creación de las características MFCC. Hay que tener en cuenta que la señal de voz es un proceso aleatorio y no estacionario, lo que supone un inconveniente a la hora de analizarla. Para salvar este problema se debe tener en cuenta que a corto plazo de tiempo (del orden de ms) la señal es casi estacionaria. Por este motivo, el primer paso consiste en dividir la señal de habla en tramas, normalmente aplicando una función de enventanado en intervalos fijos. El objetivo aquí es, por tanto, modelar pequeñas secciones de la señal que son estadísticamente estacionarias. La función de enventanado por regla general es una ventana de tipo Hamming, con un tamaño de 20 ms, que ayuda a suavizar la información de la señal original, eliminando los efectos de los bordes. Se genera un vector de características cepstrales para cada trama.

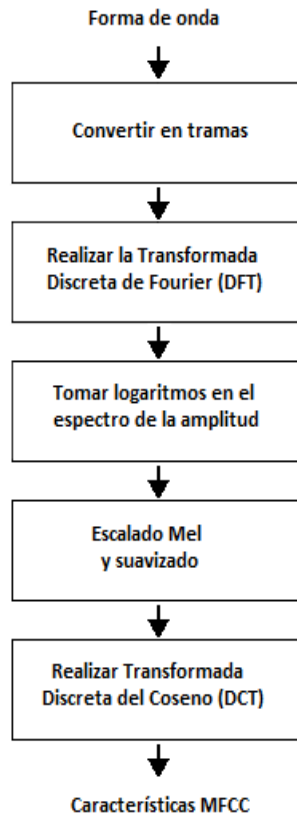


Figura 6: Proceso para crear las características MFCC

El siguiente paso es el de realizar la transformada discreta de Fourier (DFT) para cada trama.

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi nk}, 0 \leq k \leq N$$

Entonces se conserva solo el logaritmo del espectro de amplitud, descartándose la información de la fase y trabajando con la envolvente de la señal de voz $|X[k]|$. El hecho de tomar logaritmos sobre el espectro de la amplitud se debe a que se ha averiguado que el volumen percibido de la señal se puede aproximar de manera logarítmica tal y como explican en [3].

El paso siguiente consiste en suavizar el espectro y enfatizar las frecuencias más significativas. Esto se consigue recolectando los componentes espectrales en un menor número de puntos frecuenciales como se muestra en la Figura 7. Aunque uno esperaría que estos puntos estuviesen equiespaciados en frecuencia, se ha encontrado que para el habla, las frecuencias más bajas son perceptualmente más importantes que las frecuencias más altas. Por tanto, el espaciado sigue la llamada escala de frecuencias 'Mel'.

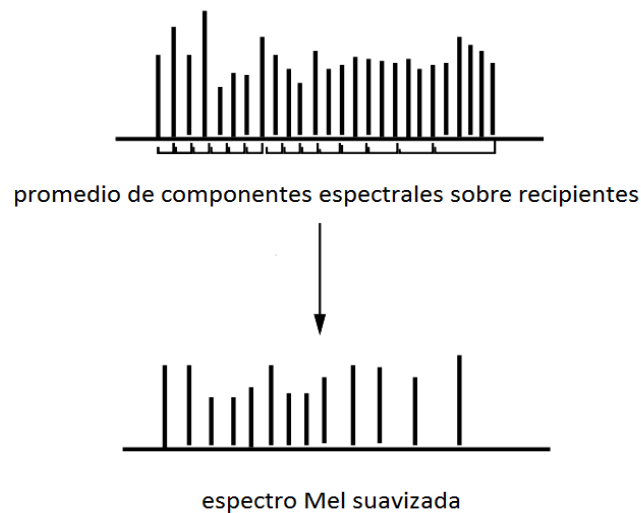


Figura 7: Escalado y suavizado de los coeficientes Mel.

La escala Mel se basa en una relación entre la frecuencia real y el tono percibido, ya que el sistema auditivo humano no funciona de manera lineal. El mapeo es aproximadamente lineal por debajo de 1kHz y logarítmico por encima. La Figura 8 muestra la función Mel.

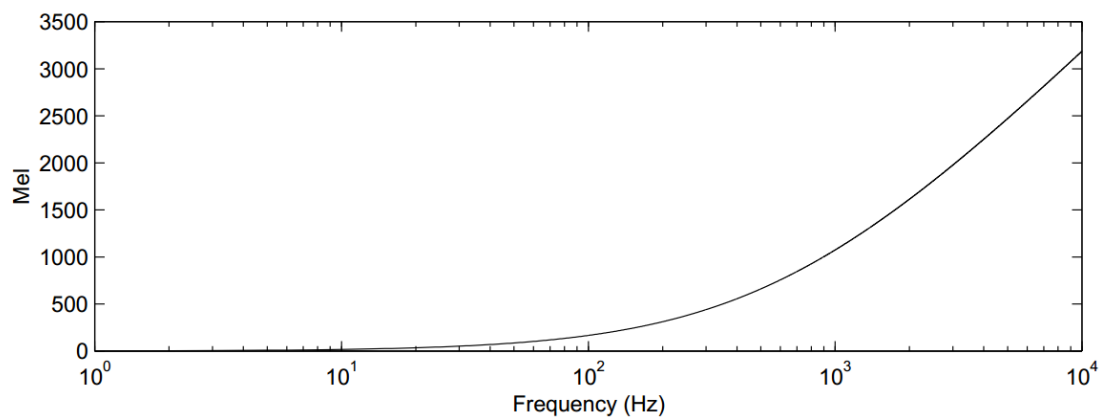


Figura 8: La escala de Mel

Los componentes de los vectores Mel-cepstral calculados para cada trama están altamente correlacionados. Las características del habla se modelan normalmente por mezclas de densidades gaussianas. Por lo tanto, con el fin de reducir el número de parámetros en el sistema, el último paso en la construcción de las características MFCC es la de aplicar una transformada a los vectores Mel-cepstral para decorrelar sus componentes. Teóricamente, la transformada de Karhunen- Loeve (KL) consigue esto, pero en tareas del habla se suele aproximar por la transformada discreta del coseno (DCT). Usando esta transformada, características cepstrales se obtienen por cada trama.

2.3.2 Modelos AR

Dada una serie temporal, sus medidas consecutivas contienen información sobre el proceso que lo generó. Un intento de describir el funcionamiento subyacente puede ser logrando modelar el actual valor de la variable como una suma lineal ponderada de los valores previos. Esto es un modelo autorregresivo (AR) y es una manera muy simple, aunque efectiva, de aproximar series temporales. El orden del modelo es el número de observaciones precedentes usadas para caracterizar la serie temporal.

Los modelos autorregresivos multivariantes (MAR) extienden esta aproximación a series temporales múltiples, de manera que el vector de valores actuales de varias variables es modelado como una suma lineal de actividades previas. Considérense d series temporales generadas a partir de d variables, y en donde m es el orden del modelo. Un modelo $MAR(m)$ predice el próximo valor de una serie temporal de dimensión d , y_n como una combinación lineal de los m vectores previos

$$y_n = \sum_{i=1}^m y_{n-i} A(i) + u_n \quad (38)$$

En donde $y_n = [y_n(1), y_n(2), \dots, y_n(d)]$ es la n -ésima muestra de la serie temporal d -dimensional, cada $A(i)$ es una matriz $d \times d$ de pesos y $u_n = [u_n(1), u_n(2), \dots, u_n(d)]$ es ruido Gaussiano aditivo con media cero y covarianza R , habiendo asumido que la media de los datos ha sido sustraída de la serie temporal.

El modelo puede ser reescrito de una manera estándar como un modelo de regresión lineal multivariable de la siguiente manera

$$y_n = x_n W + u_n \quad (39)$$

en donde $x_n = [y_{n-1} \ y_{n-2} \ \dots \ y_{n-m}]$ son los m vectores multivariantes previos y W es una matriz $(m \times d) \times d$ con todos los pesos MAR. Por tanto, habrá un total de $m \times d \times d$ pesos MAR. Puede consultarse más información sobre los modelos MAR en [2].

CAPÍTULO 3 DISEÑO E IMPLEMENTACIÓN DE LOS CLASIFICADORES

3.1 Introducción

En este proyecto se pretende comprobar si es posible averiguar de qué está hablando un narrador teniendo en cuenta únicamente el tono con el que habla, el “cómo” lo dice, y no las palabras que utiliza, es decir, el “qué” está diciendo. Para ello se han diseñado e implementado tres tipos de clasificadores distintos haciendo uso de técnicas estándar utilizadas habitualmente en la clasificación de géneros musicales. En este capítulo se describe la implementación de los clasificadores descritos en el Capítulo 2.

3.2 Implementación: MATLAB

Para la implementación de dichos clasificadores se ha utilizado el software matemático Matlab, que es la abreviatura de MATrix LABoratory (laboratorio de matrices). *MATLAB* es un entorno de cálculo técnico de altas prestaciones que integra análisis numérico, cálculo matricial, procesamiento de señales y visualización gráfica en un entorno completo donde los problemas y sus soluciones son expresados del mismo modo en que se escribirían.

El motivo por el que se ha elegido MATLAB es porque dispone de un lenguaje propio (lenguaje M) que es sencillo, intuitivo y trabaja muy bien con vectores y matrices.

Otra de las razones por las que se ha utilizado este software matemático es la posibilidad de integrar la librería LIBSVM para poder trabajar de manera fácil y compatible con las Máquinas de Soporte Vectorial.

A lo largo de este capítulo se irán indicando las funciones y librerías especiales utilizadas para llevar a cabo la implementación de los clasificadores.

3.3 Diseño

El diagrama de bloques que se expone a continuación detalla los pasos necesarios para el diseño de los tres clasificadores:

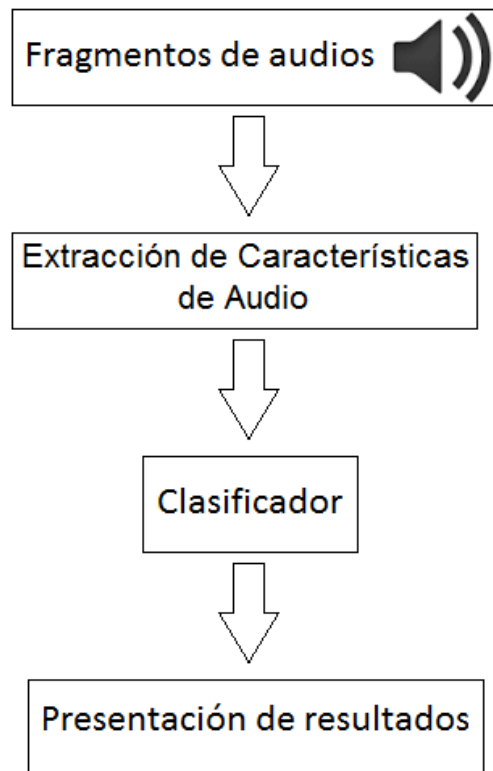


Figura 9. Diseño clasificador

En los siguientes puntos se desarrollan los distintos bloques de la Figura 9.

3.4 Fragmentos de audios

Para poder probar los clasificadores diseñados es necesario un conjunto de audios. Para este proyecto en concreto se utilizará una base de datos que será detallada en el Capítulo 4 (apartado 4.2). Los audios en formato mp3 se guardan en un archivo con extensión .mat y se cargan en matlab mediante el comando "load".

Para crear el archivo .mat con los audios cargados se ha tenido que hacer uso previamente de una función MFCC_MAR.m, en la que se pasa por parámetro el “path” de la carpeta donde se encuentran los audios guardados, y en la cuál se extraerán las características relevantes de los mismos, tal y como se verá con más detenimiento en el apartado siguiente. De esta manera, si queremos hacer uso de los clasificadores diseñados en este proyecto para un conjunto de audios distinto, únicamente deberíamos cambiar dicho directorio indicando donde se encuentran los nuevos audios a clasificar, y después volver a crear un nuevo archivo con extensión .mat para poder ser cargado en Matlab.

3.5 Extracción de Características de Audio

Una vez cargados los audios, se extraen las características relevantes de los mismos para poder llevar a cabo la tarea de clasificación automática. Para ello se usarán los coeficientes MFCC, que tal y como se indicó en el Capítulo 2, son utilizados para la representación del habla y se basan en la percepción auditiva humana, conteniendo información sobre las variaciones de la curva espectral de la señal de audio. Gracias a los MFCC se consigue extraer características de los componentes de una señal de audio que sean adecuadas para la identificación de contenido relevante, así como obviar todas aquellas que posean información poco valiosa como el ruido de fondo, emociones, volumen, tono... que no aportan nada al proceso de reconocimiento.

La técnica utilizada en este proyecto es la descrita en [4]. Primero se extraen los MFCCs en ventanas de poca duración que se solapan entre sí. El tamaño de la ventana se fija en 30 ms y el del salto en 15 ms. Se usa un banco con 30 filtros, y solo se mantienen los 6 coeficientes iniciales (descartándose el primer coeficiente, asociado a la dimensión perceptual del nivel sonoro).

Se sabe que el uso directo de MFCCs no proporciona una adecuada representación para llevar a cabo la tarea de clasificación, es por esto que posteriormente a la extracción se realizará un proceso de integración en el tiempo con el objetivo de recuperar información más relevante.

Como alternativa a procedimientos más simples, tales como el uso de la media y varianza de los MFCCs, en el artículo [2] se propuso ajustar un modelo autorregresivo multivariante (*Multivariate Autorregresive*, MAR). Es decir, para un conjunto de vectores MFCCs consecutivos, se ajusta un modelo MAR de intervalos P usando la fórmula, $x_n = \sum_{p=1}^P A_p x_{n-I(p)} + u_n$ en donde x_n son los MFCCs extraídos en la ventana n -ésima, u_n es el error de reconstrucción y A_p las matrices de coeficientes del modelo autorregresivos multivariante de orden P . Los valores de las matrices A_p , $p = 1, \dots, P$, junto con la media y la covarianza del ruido u_n se concatenan en un vector de características (vector MAR).

Los modelos MAR usados son de orden $P = 3$, siendo los vectores MAR de un tamaño igual a 135 (Figura 10). Para esta fase de integración en el tiempo, se han considerado un tamaño de ventana de 2 s y un tamaño de salto de 1 s.

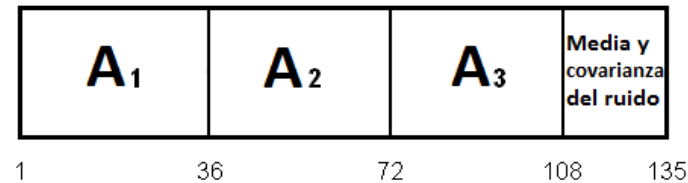


Figura 10. Vector MAR

Las características MFCC extraídas serán las observaciones que utilizaremos para llevar a cabo la tarea de la clasificación. En este trabajo se ha decidido probar tres clasificadores diferentes, cuyo diseño e implementación serán explicados en los puntos siguientes.

3.6 Clasificadores

3.6.1 Máxima Verosimilitud (ML)

Las observaciones obtenidas anteriormente serán los datos que se usarán en la tarea de la clasificación, siendo utilizados la mitad para entrenar y la otra mitad para test. El motivo por el que se realiza esta partición es para poder evaluar la capacidad de generalización del clasificador, es decir, que sea capaz de clasificar muestras que no fueron usadas en la etapa de entrenamiento.

Tal y como se detalló en el Capítulo 2, el modelo que se va a utilizar está basado en Gaussianas, una por cada una de las clases elegidas para este trabajo. Por tanto, tenemos que aprender las medias y las covarianzas de los datos de entrenamiento. De esta manera se obtendrán dos matrices por clase: una que contendrá las medias con dimensión 1×135 y otra las covarianzas con dimensión 135×135 .

Las medias han sido calculadas mediante la función de matlab *mean*. Se calcularon tres covarianzas distintas, las cuales serán denominadas covarianza **Identidad**, covarianza **Diagonal** y covarianza **Multidimensional**. En el capítulo 4, apartado 4.3.1 se explicará el motivo de estas tres covarianzas, las cuales fueron implementadas del siguiente modo:

-Covarianza Identidad:

```
covarianza=var_identidad1(Xtr);
```

Siendo *var_identidad1.m* implementada de la siguiente forma:

```
function v = var_identidad1(x)
    sigma=mean(var(x));
    v=sigma*eye(size(x,2));
```

-Covarianza Diagonal:

```
covarianza=diag(var(Xtr);
```

-Covarianza Multidimensional:

```
covarianza=cov(Xtr);
```

Una vez aprendidas la media y la covarianza, se toman los datos de test y se evalúa la verosimilitud de cada muestra en cada una de las Gaussianas. Para ello se llama a la función *calculaGaussiana.m*, y se le pasan como argumentos los datos de test a clasificar (teniendo en cuenta que constan de 135 elementos) y la media y la covarianza calculadas, de la siguiente manera:

```
gaussianas=calculaGaussiana(Xtst(:,n),media,covarianza);
```

Se obtendrán cuatro matrices (una por cada clase) de cuatro columnas y tantas filas como trozos tengan los datos de test utilizados. Cada columna corresponderá a una clase diferente. Para poder realizar la clasificación se debe ver qué clase se corresponde con la gaussiana donde se alcanza la máxima verosimilitud por parte de cada dato.

Los errores, y por tanto, los aciertos, se contabilizarán de dos formas distintas, por audio completo o por trozo de audio, tal y como se explicará en el Capítulo 4, apartado 4.2.1.

3.6.2 Vecino más próximo (KNN)

Para realizar la clasificación haciendo uso del algoritmo del Vecino más próximo, lo primero que se debe calcular es la K óptima mediante una técnica de validación cruzada utilizando para ello únicamente los datos de entrenamiento. Esta K indicará el número de vecinos necesarios para llevar a cabo la tarea de la clasificación.

El siguiente paso será el de hallar la **distancia euclídea** de todos los datos de test con respecto a los datos de entrenamiento, obteniéndose una matriz con las distancias de todos los trozos entre sí. La distancia euclídea entre dos puntos P_1 y P_2 , de coordenadas cartesianas (x_1, y_1) y (x_2, y_2) se define así:

$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Una vez hallada la matriz de distancias, se ordenará cada fila de menor a mayor (utilizando el comando *sort* de matlab, opción “*ascend*”), y se cogerán tantas posiciones como indique la K calculada. Posteriormente, teniendo en cuenta las etiquetas de los K vecinos seleccionados, asignaremos al nuevo dato la etiqueta de la clase mayoritaria. Tal y como se vio en el Capítulo 2, al disponer de cuatro clases es importante que la K óptima sea un número impar para evitar posibles empates a la hora de decidir.

Si la etiqueta impuesta al nuevo dato coincide con la etiqueta real, se apunta un acierto, si no coincide se apunta un fallo con respecto a la clase de la etiqueta asignada, para poder ver con qué clases se confunden más y de cuáles se diferencian mejor.

3.6.3 Máquinas de Vectores de Soporte (SVM)

3.6.3.1 Librería LIBSVM

LIBSVM fue creada en 2001 por Chih-Chung Chang y Chih-Jen Lin [5]. Es una librería software para máquinas de soporte vectorial que integra clasificación de vectores soporte (C-SVC, nu-SVC), regresión (épsilon-SVR, y nu-SVR), estimación de distribuciones y clasificación multiclase.

Aparte de todas las características mencionadas en el párrafo anterior, lo que hace a LIBSVM verdaderamente interesante es, como se comentó en el apartado 3.1, la posibilidad de integrarla bajo el software matemático MATLAB.

De esta librería se han utilizado dos funciones:

- ❖ **svmtrain.** Realiza el entrenamiento de unos datos a partir de unos parámetros. Esta función se utiliza de la siguiente manera:

```
model = svmtrain(Ytr, Xtr, options);
```

Donde *Xtr* serán los datos de entrenamiento, e *Ytr* sus etiquetas. La variable *options* se configurará de distinta forma en función del tipo de kernel a utilizar. Aquí se comentan los más relevantes:

- **-s svm_type:** Selecciona el tipo de SVM: 0 para SVM de clasificación y 3 para SVM de regresión.
- **-t kernel_type:** Para seleccionar el Kernel: 0 para lineal, 1 para polinomial, 2 para RBF y 3 para sigmodial.
- **-c cost:** establece el parámetro C de C-SVC. Indica cuanto queremos ajustarnos al conjunto de entrenamiento. Es la C del problema de optimización.
- **-g gamma:** asigna el valor del parámetro a la función Kernel.

El valor de los parámetros C y gamma óptimos, es decir, con menor porcentaje de error, se hallará mediante la técnica de validación cruzada. El cálculo de los mismos podrá verse en el Capítulo 4, apartado 4.3.3.1.

La función devuelve *model*, que es un conjunto de datos con los resultados del entrenamiento.

- ❖ **svmpredict.** Esta otra función clasifica un conjunto de muestras una vez realizado el entrenamiento:

```
[pred1, accuracy, decision_values] = svmpredict(Ytst, Xtst, model);
```

La función recibe tres argumentos: las etiquetas de los datos de test (*Ytst*, para contar los datos automáticamente), los datos de test a clasificar *Xtst* y el modelo obtenido en el entrenamiento (*model*). Devuelve la predicción de la clase para cada muestra, la precisión y la salida blanda. El uso de estas dos funciones externas se debe a la necesidad de contar con una clasificación a priori, *decision_values*, la cual será utilizada en el clasificador a posteriori.

En la página web de LIBSVM (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>) se pueden adquirir distintos manuales y el software necesario para diferentes entornos.

3.6.3.2 Diseño e implementación SVM

Para poder hacer uso de la librería LIBSVM hay que seguir una serie de pasos. En primer lugar hay que adjuntarla al path de Matlab, posteriormente hay que ejecutar el comando *mex -setup*, y seleccionar un compilador. Para la versión de Matlab disponible se utilizó Microsoft Visual C++ 2012 (Microsoft Visual Studio 11.0). Por último, ejecutar el comando *make*. Una vez realizados estos pasos, ya es posible utilizar las funciones mencionadas en el punto 3.6.3.1 en Matlab.

La LIBSVM es una implementación de las SVM muy rápida y robusta. Esta implementación es capaz de trabajar con más de dos clases, por lo que soluciona el problema de que la SVM sea un clasificador binario: las clases no serán +1 y -1, sino que serán números enteros consecutivos del 1 en adelante. Aplicado al caso que nos ocupa en este proyecto, al querer clasificar 4 clases diferentes, estas serán 1, 2, 3 y 4. El algoritmo que usa la LIBSVM para la clasificación multiclase es el llamado “Uno-contra-uno” visto en el Capítulo 2, apartado 2.2.4.

Validación cruzada

Se ha realizado el diseño de las SVM para dos tipos de kernels: lineal y gaussiano. Para el kernel lineal se debe calcular la C óptima y para el gaussiano la C y la gamma óptimas. Ambos, como se comentó anteriormente, serán calculados mediante la técnica de validación cruzada. Para ello se utilizarán únicamente los datos de entrenamiento y sus etiquetas correspondientes, en función de la clase a la que pertenecen: 1, 2, 3 o 4.

En primer lugar se definen los N candidatos, es decir, el conjunto de C entre los cuales se va a decidir el más adecuado para usar en el entrenamiento por ser el que obtiene menor porcentaje de error. En este trabajo, N=6, por lo que se proponen los siguientes candidatos: 0.01, 0.1, 1, 10, 100, 1000 tanto para el kernel lineal como para el gaussiano. Por otro lado se definen también para este último las gammas:

$$\frac{1}{16}\sqrt{135}, \frac{1}{8}\sqrt{135}, \frac{1}{4}\sqrt{135}, \frac{1}{2}\sqrt{135}, \sqrt{135}.$$

Después se divide el conjunto de datos de entrenamiento en 5 subconjuntos y se selecciona el primero de esos subconjuntos como conjunto de validación (XV) y el resto como conjunto de entrenamiento (XR), teniéndose a su vez el conjunto de validación de las etiquetas (YV) y el de entrenamiento (YR). Se realizan los cálculos con esos conjuntos y se almacena el resultado. Posteriormente se coge el segundo subconjunto como conjunto de validación y el resto como conjunto de entrenamiento, y así sucesivamente, realizando los cálculos en 5 ocasiones, obteniendo la media de los cinco resultados. (Ver Capítulo 4, apartado 4.3.3).

Primero se configura la variable options:

- Para el kernel lineal:

```
options=['-t 0 -s 0 -c ' num2str(C)];
```

-t 0 por ser lineal

-s 0 por ser un SVM de clasificación

-c irán recorriendo los distintos valores definidos de C

- Para el kernel gaussiano:

```
options=['-t 2 -s 0 -c ' num2str(C) ' -g ' num2str(G)];
```

-t 2, por ser RBF, o gaussiana

-s 0 por ser un SVM de clasificación

-c irán recorriendo los distintos valores definidos de C

-g irán recorriendo los distintos valores definidos de gamma

Con la función **svmtrain** se halla el modelo pasando como argumentos los conjuntos de entrenamiento y sus etiquetas y las opciones seleccionadas anteriormente:

```
model = svmtrain(YR, XR, options);
```

Y por último, con la función **svmpredict** se obtienen los valores predichos pasando como argumentos el conjunto de validación y sus etiquetas, y el modelo definido anteriormente con la función svmtrain:

```
[pred1, accuracy, decision_values] = svmpredict(Yv, Xv, model);
```

Se comparará el número que se obtiene en pred1 con Yv, si es diferente se contabiliza un error. Una vez calculados todos los errores, se elegirán los valores de C y gamma que menos porcentaje de error hayan obtenido.

Entrenamiento y clasificación

Una vez calculadas las constantes anteriores mediante validación cruzada se realizará la clasificación haciendo uso de las mismas funciones de la librería LIBSVM. En primer lugar se cargarán los datos de entrenamiento y test y se asignarán sus etiquetas correspondientes.

Se eligen las opciones con los valores obtenidos en el punto anterior. Para nuestro caso particular, los valores óptimos de C y gamma obtenidos serán los siguientes:

Kernel lineal: `options=['-t 0 -s 0 -c 1'];`

Kernel gaussiano: `options=['-t 2 -s 0 -c 1 -g (1/16*sqrt(135))'];`

Entrenamiento:

Utilizando la función **svmtrain**, se pasará como argumentos los datos de entrenamiento (Xtr), las etiquetas de los mismos con las clases a las que pertenecen (Ytr) y la cadena de texto *options* detallada anteriormente.

```
model = svmtrain(Ytr, Xtr, options);
```

Esta función devuelve un modelo que es una estructura de datos que recoge los parámetros que definen el clasificador, por lo que será utilizado en el siguiente paso, es decir, en las pruebas del clasificador.

Clasificación:

La clasificación se lleva a cabo utilizando la función de LIBSVM **svmpredict**. Haciendo uso del modelo obtenido de la llamada a la función de entrenamiento *svmtrain*, se realiza la llamada a *svmpredict*, junto los datos de test (Xtst) y sus etiquetas (Ytst).

```
[pred1, accuracy, decision_values] = svmpredict(Ytst, Xtst, model);
```

A su salida, nos encontramos con las etiquetas de la clasificación, *pred1*, que servirán para evaluar el comportamiento del clasificador. Si la etiqueta *pred1* coincide con la etiqueta *Ytst* anotaremos acierto, en caso contrario se comprobará con cual de las clases restantes coincide para apuntar el fallo.

3.7 Presentación de resultados

Para presentar los resultados se han utilizado matrices de confusión o también llamadas de clasificación. Una matriz de confusión es una herramienta de visualización que se emplea en el aprendizaje supervisado. Cada fila representa las clases a clasificar y cada columna el número de predicciones de cada clase, por lo que es fácil ver tanto los aciertos como los fallos cometidos. Interesa que los valores altos se encuentren en los elementos de la diagonal principal, ya que estos serían los aciertos, y el resto de elementos sean números bajos, puesto que son los errores. Gracias a esta matriz se podrán ver con facilidad que clases se confunden más.

CAPÍTULO 4 EXPERIMENTOS Y RESULTADOS

4.1 Introducción

En este capítulo se muestran los resultados obtenidos en los diferentes experimentos realizados en forma de matrices de confusión. Previamente se presentará la base de datos utilizada y se ofrecerá una breve descripción de los experimentos llevados a cabo. Después se mostrarán los resultados conseguidos así como comentarios acerca de los mismos, y por último se realizará una comparativa de los resultados de los tres clasificadores elegidos para este trabajo.

4.2 Base de datos

Para construir la base de datos se han utilizado audios podcast descargados en formato MP3 de los espacios web www.ivoox.com, www.rtve.es/radio/podcast y www.cope.es/Podcast-Tiempo-de-juego. Los audios pertenecen a diferentes estilos de narración en **español**: *Cuentos*, *Noticias*, *Fútbol* y *Monólogos*, en donde aparece un único hablante de género **masculino**.

De los podcast indicados anteriormente, se han extraído fragmentos de audio de 30 segundos de duración y se han eliminado los silencios largos entre las palabras pronunciadas por el locutor. Para ello se ha utilizado un programa llamado *Free MP3 Cutter and Editor*.

Como los clasificadores funcionan en formato mono, se han transformado aquellos audios que habían sido grabados en estéreo haciendo uso del programa *Audacity*.

Teniendo en cuenta que la clasificación va a basarse únicamente en una onda sonora que genera la voz del hablante, y al no disponer de instrumentos que puedan ayudar a identificar una clase u otra (como en la clasificación por género musical), se ha tenido especial cuidado en seleccionar aquellos audios que fueran más característicos para facilitar el trabajo del clasificador. Por ejemplo, en la clase Fútbol se han cogido fragmentos de audio en los que el locutor estaba radiando un gol o una jugada y ponía más énfasis a la hora de hablar. En la clase Cuentos, en cambio, el tono es más sosegado al dirigirse a un público infantil. En la clase Monólogos, el tono es más distendido y coloquial, adaptándose la voz al discurso del hablante, y en la clase Noticias, el tono es más formal y uniforme.

La base de datos estará formada por 20 fragmentos de audios de cada estilo de narración: Cuentos, Noticias, Fútbol y Monólogos, es decir, se dispondrá de un conjunto de 80 fragmentos de audios, siendo cada uno de ellos de 30 segundos de duración.

4.3 Experimentos realizados

A continuación se explicarán los distintos experimentos que se han llevado a cabo en este proyecto. Estos experimentos se basan principalmente en ir combinando los fragmentos de audios de la base de datos formando distintos conjuntos de entrenamiento y test que serán utilizados posteriormente en los clasificadores desarrollados. Hay que tener en cuenta que, como se vio en el diseño del clasificador, cada audio se compone de trozos, por lo que dichos trozos a su vez podrán ser combinados constituyendo nuevos y diferentes conjuntos de entrenamiento y test.

4.3.1 Experimento 1

En este primer experimento se utilizarán la mitad de los audios de cada clase para entrenar y la otra mitad para testear: Se han dividido los 80 audios que forman la base de datos en dos grupos de 40 audios. Uno formado por 10 audios de cada una de las clases: Cuentos, Noticias, Fútbol y Monólogos, y el otro por los 40 audios restantes (10 audios también de cada clase). De estos audios han sido extraídos los datos para construir los conjuntos: El conjunto de entrenamiento está constituido por los trozos de uno de los dos grupos formado por 40 audios y el conjunto de test con el otro. Se han realizado 10 combinaciones diferentes de los 80 audios para hacer 10 repeticiones de este experimento y poder obtener resultados con validez estadística.

Una vez delimitados ambos conjuntos, se han realizado a su vez distintos tipos de clasificación:

- I. Clasificación por **trozos**: se cogen las salidas que ha predicho el clasificador para cada uno de los trozos individuales, obteniéndose así los porcentajes de acierto y de error .
- II. Clasificación por **audios**: esta vez se cogen las salidas que ha predicho el clasificador para todos los trozos de un mismo audio. Se comprueba cual es la moda (es decir, el valor que más se repite) y esa sería la predicción para el género del audio completo. Si coincide con el del género se anotaría un acierto y en caso contrario, un fallo. Finalmente se contabilizarían los resultados para saber el porcentaje de acierto y error que existe por audios completos.
- III. Por último se han repetido los apartados I y II utilizando únicamente el conjunto de datos de entrenamiento tanto para entrenar como para test, para poder comprobar si se está sobreajustando o no.

4.3.2 Experimento 2

En este experimento lo que se ha hecho ha sido dividir cada audio en dos y utilizar la primera mitad como datos de test y la segunda mitad como datos de entrenamiento. Con esto se pretende cuantificar la mejora obtenida en los resultados al enriquecer la base de datos, especialmente en las clases que se confunden peor, al estar formados los datos de entrenamiento por las mitades de todos los audios. Por otro lado cabe comentarse que este experimento solo se ha realizado una vez. La razón de que solo se haya ejecutado una repetición es que a diferencia del apartado anterior en el que se realizaban 10 combinaciones diferentes de los audios para formar los datos de entrenamiento y test, en este experimento no hay más combinaciones posibles.

4.3.3 Experimento 3

En este apartado se repetirá el experimento anterior, pero esta vez, en vez de utilizar la primera mitad para entrenar y la segunda para testear, se va a dividir nuestro audio en los trozos que lo forman y se cogerán de forma aleatoria la mitad de los trozos para formar los datos de entrenamiento y la mitad restante para formar los datos de test. Este experimento, se realizará también en 10 ocasiones cambiando los trozos que forman cada uno de los conjuntos de datos para obtener un resultado robusto estadísticamente hablando, por lo que después se promediarán los resultados obtenidos.

4.3.4 Experimento 4

En este experimento lo que se va a realizar es dejar cada vez una clase fuera del conjunto de entrenamiento. Lo que va a interesar principalmente es comparar la fila de la clase que se ha dejado fuera para ver como se reparte entre las otras tres clases, si se reparte equitativamente o los trozos caen más en una clase que en otra. Esta clasificación se ha realizado tanto por trozos como por audios para poder comparar los resultados.

4.4 Resultados obtenidos

En este apartado se incluirán todos los resultados obtenidos en los cuatro experimentos detallados en el apartado anterior para cada uno de los clasificadores y se realizarán comentarios acerca de los mismos.

Para representar los resultados se han utilizado matrices de confusión o también llamadas de clasificación. Una matriz de confusión es una herramienta de visualización que se emplea en el aprendizaje supervisado. Cada fila representa las clases a clasificar y cada columna el número de predicciones de cada clase, por lo que es fácil ver tanto los aciertos como los fallos cometidos. Interesa que los valores altos se encuentren en los elementos de la diagonal principal, ya que estos serán los aciertos, y el resto de elementos sean números bajos, puesto que son los errores. Gracias a esta matriz se podrán ver con facilidad que clases se confunden más.

Todos los resultados de los experimentos que se incluyen a continuación han sido repetidos utilizando únicamente los datos de entrenamiento tanto para test como para entrenar, de tal manera que se ha podido comprobar que el sistema no sobreajusta. Estos resultados pueden consultarse en el Apéndice.

4.4.1 Resultados clasificador de Máxima Verosimilitud

En el clasificador ML, se utiliza la fórmula de la Gaussiana Multidimensional que se define de la siguiente manera tal y como se vio en el Capítulo 2 (apartado 2.5).

$$f_{\underline{x}}(\underline{x}) = G_{\underline{x}}(\underline{m}, \underline{V}) = \left[(2\pi)^N |\underline{V}| \right]^{-1/2} \cdot e^{-\frac{1}{2}(\underline{x}-\underline{m})^T \underline{V}^{-1}(\underline{x}-\underline{m})}$$

Para poder hacer uso de esta fórmula hay que calcular la media y la covarianza de los datos de entrenamiento. Ya que hay distintas posibilidades para calcular la covarianza, se ha querido realizar los experimentos de tres formas diferentes modificando la matriz de covarianzas (\underline{v}) para poder comparar los resultados y decidir con cuál de las matrices el porcentaje de error es menor. A partir de ahora llamaremos a los detectores de tres maneras: Identidad, Diagonal y Multidimensional en función de la covarianza utilizada:

- **Detector Identidad:** Si se piensa que el conjunto de datos de entrenamiento \underline{x} viene de una gaussiana con matriz de covarianzas igual a la identidad por una constante.
- **Detector Diagonal:** Si se cree que el conjunto de datos de entrenamiento \underline{x} viene de una gaussiana con matriz de covarianzas diagonal.

- **Detector Multidimensional:** Por último, si se piensa que el conjunto de datos de entrenamiento x viene de una gaussiana multidimensional con matriz de covarianzas completa.

4.4.1.1 Experimento 1

Se han realizado cálculos con los tres detectores para poder comparar cuál de los tres acierta más y poder elegir de qué gaussiana es más probable que provengan los datos de entrenamiento. Primero se comentarán los resultados obtenidos de cada detector por separado, tanto por trozos como por audios completos y seguidamente, en base a estos resultados, se realizará una comparativa de los tres decisores.

Tal y como se ha comentado previamente, los datos que se ponen a continuación son la media y la desviación típica de las 10 repeticiones efectuadas (media \pm desviación típica).

DETECTOR IDENTIDAD:

| Trozos/Test Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------|-------------------------------------|-----------------------------------|------------------------------------|------------------------------------|
| Cuentos | 74,04 \pm 11,31 | 2,61 \pm 1,55 | 17,3 \pm 13,19 | 6,05 \pm 2,81 |
| Noticias | 1,13 \pm 0,49 | 88,4 \pm 4,75 | 2,73 \pm 1,87 | 7,74 \pm 3,34 |
| Fútbol | 15,51 \pm 8,21 | 7,23 \pm 5,81 | 56,63 \pm 7,77 | 20,63 \pm 7,06 |
| Monólogos | 8,92 \pm 2,84 | 7,18 \pm 2,2 | 10,05 \pm 3,43 | 73,85 \pm 5,37 |

Matriz de confusión 1.1.1.a

En la Matriz de confusión 1.1.1.a se puede ver como los valores más elevados están en la diagonal principal, por lo que se puede afirmar que el detector consigue clasificar bien las cuatro clases. Ahora bien, la clase **Noticias** es la que más tasa de aciertos tiene (88,4%) detrás le siguen la clase **Cuentos** (74,04%) y **Monólogos** (73,85%). En último lugar se encuentra la clase **Fútbol** con un 56,63%, que aunque consigue clasificar más de la mitad de las muestras, se confunde en un 15,51% con la clase **Cuentos** y en un 20,63% con la clase **Monólogos**.

Observando las desviaciones típicas, llama la atención sobretodo la fila de la clase **Cuentos**, donde hay una desviación del 11,31% con respecto a los aciertos y un 13,19% con respecto a los errores cometidos al clasificar los trozos como si fueran de la clase **Fútbol**. Aunque en media el resultado era bastante aceptable, hay que tener en cuenta estas desviaciones para poder comparar correctamente los resultados con el resto de decisores.

| Audios/Test Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------|----------------|------------|----------------|---------------|
| Cuentos | 78 ± 25 | 0 | 22 ± 25 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 6 ± 5 | 10 ± 14 | 71 ± 11 | 13 ± 14 |
| Monólogos | 1 ± 3 | 0 | 1 ± 3 | 98 ± 4 |

Matriz de confusión 1.1.2.a

Los resultados sin embargo mejoran al clasificar por audios completos. En la Matriz de confusión 1.1.2.a vemos que **Noticias** continúa siendo la clase con mejores resultados, en este caso un 100% de los audios es clasificado de forma correcta. En **Monólogos** hay una mejoría notable puesto que se consigue clasificar bien el 98% de los audios, obteniéndose un 32,7% más de aciertos que en la clasificación por trozos. La clase **Cuentos** también mejora, aunque más discretamente (obtiene un 5,3% más de aciertos), pero sigue destacando el porcentaje de errores ya que un 22% de audios se siguen confundiendo con la clase **Fútbol**. Por último, es interesante ver como mejoran los resultados en la clase **Fútbol** un 25% en comparación con la clasificación por trozos, ya que Fútbol era la clase la que más errores cometía.

Observando las desviaciones típicas destacan sobretudo los valores obtenidos en la clase **Cuentos** con una variabilidad del 25% tanto en aciertos como en errores al clasificar los audios como si fueran de la clase **Fútbol**, al igual que pasó en la clasificación por trozos. También llama la atención los valores obtenidos en la clase Fútbol, tanto en aciertos (11% de desviación) como en errores (14% en la clase Noticias y Monólogos).

DETECTOR DIAGONAL:

| Trozos/Test Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 83,26 ± 8,55 | 3,54 ± 1,97 | 6,68 ± 6,91 | 6,52 ± 3,82 |
| Noticias | 1,42 ± 1,06 | 95,51 ± 5,89 | 1,21 ± 2,35 | 1,86 ± 2,76 |
| Fútbol | 7,69 ± 4,38 | 2,00 ± 1,2 | 77,09 ± 9,11 | 13,22 ± 8,47 |
| Monólogos | 1,61 ± 0,97 | 6,13 ± 2,85 | 12,51 ± 5,77 | 79,75 ± 8,88 |

Matriz de confusión 1.1.3.a

En la Matriz de confusión 1.1.3.a podemos ver como mejoran los resultados de las cuatro clases con respecto al detector Identidad. La clase **Noticias** continúa siendo la que tiene un porcentaje de acierto más elevado con un 95,51 % (mejora un 8,04%) y la clase **Fútbol** la que lo tiene más bajo, aunque ha mejorado un 36,12% en media con respecto al detector Identidad. La clase **Cuentos** obtiene una mejora del 12,45% y la clase **Monólogos** del 7,98%. Como curiosidad podemos destacar como tanto la clase **Fútbol** como **Monólogos** tienen una tasa de acierto bastante similar (77,09% y 79,75% respectivamente) y que a su vez tienen una tasa de fallo con respecto a la otra clase también parecida, ya que un 13,22%

de los datos de la clase **Fútbol** son clasificados como **Monólogos** y un 12,51% de la clase **Monólogos** son clasificados como **Fútbol**.

Observando las desviaciones típicas y comparando los resultados con los obtenidos en el decisor Identidad, podemos ver como las dispersiones de todas las clases, a excepción de la clase **Cuentos**, son mayores que las obtenidas en la Matriz de confusión 1.1.1.a.

Teniendo en cuenta tanto las dispersiones obtenidas en el decisor Identidad como Diagonal podemos ver como algunos resultados solapan entre sí. A pesar de esto podemos seguir afirmando que con el decisor Diagonal se obtienen mejores resultados, ya que en media los porcentajes de acierto son más altos, aunque esta mejora se relativiza teniendo en cuenta las desviaciones.

| Audios/Test Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------|----------------|---------------|----------------|---------------|
| Cuentos | 90 ± 16 | 0 | 8 ± 13 | 2 ± 4 |
| Noticias | 0 | 99 ± 3 | 0 | 1 ± 3 |
| Fútbol | 5 ± 5 | 1 ± 3 | 88 ± 11 | 6 ± 10 |
| Monólogos | 0 | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 1.1.4.a

En la Matriz de confusión 1.1.4.a podemos ver como mejoran los porcentajes al clasificar por audios completos en vez de por trozos, como se mostraba en la matriz 1.1.3.a. Las clases **Noticias** y **Monólogos** son las que obtienen mejores resultados, puesto que logran un 99% de acierto. Siendo **Monólogos** la que más destaca con un 24,13% más de aciertos que en la clasificación por trozos. La clase **Noticias** mejora un 3,65% y la clase **Cuentos** un 8,09%. La clase **Fútbol** vuelve a obtener el porcentaje menor, aunque consigue un 14,15% más de acierto.

Comparando con la clasificación por audios del decisor Identidad (Matriz 1.1.2.a), mejoran todas las clases excepto la clase **Noticias**, donde nos aparece un porcentaje ligeramente inferior (99%) y un 1% es clasificado como **Monólogos**. Esto puede llamar la atención puesto que el porcentaje de aciertos para los trozos había aumentado con respecto al detector Identidad, pero en cambio por audios este se ve decrementado, aunque en un porcentaje de error tan pequeño que sería prácticamente despreciable. Con respecto al resto de clases, todas mejoran notablemente: la clase **Cuentos** un 15,38%, **Fútbol** un 23,94 % y **Monólogos** un 1,02%.

Observando la Matriz de confusión 1.1.4.a podemos ver como el grado de dispersión ha disminuido, sobretodo en la clase **Cuentos**, donde antes teníamos un 25% y ahora un 16% de dispersión. Por lo que podemos afirmar que los resultados superan los hallados en el decisor anterior, no solo porque las medias obtenidas sean mejores, sino por que además las desviaciones están más próximas a dichas medias.

DETECTOR MULTIDIMENSIONAL:

| Trozos/Test Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 85,78 ± 6,52 | 4,71 ± 3,03 | 2,78 ± 2,06 | 6,73 ± 4,43 |
| Noticias | 2,35 ± 1,81 | 95,41 ± 2,33 | 0,91 ± 0,77 | 1,33 ± 0,62 |
| Fútbol | 2,77 ± 2,09 | 0,59 ± 0,68 | 79,35 ± 7,51 | 17,29 ± 6,53 |
| Monólogos | 1,19 ± 0,76 | 0,31 ± 0,26 | 5,97 ± 3,09 | 92,53 ± 3,09 |

Matriz de confusión 1.1.5.a

En la Matriz de confusión del detector Multidimensional vemos como globalmente mejoran los resultados con respecto a los dos detectores anteriores. La clase **Noticias** persiste siendo la que obtiene más aciertos, un 95,41 %, es decir, un 0,1% menos que en el detector Diagonal pero un 7,92% más que el detector Identidad. Por otro lado, la clase **Fútbol** continúa siendo la que tiene más problemas para clasificarse por trozos. El detector Multidimensional obtiene para esta clase un 2,93% más de aciertos que el Diagonal y un 40,12% más que el Identidad. Destaca como un 17,29% de los trozos de esta clase sigue cayendo sobre la clase **Monólogos**.

Observando las soluciones de las otras dos clases restantes podemos ver como la clase **Cuentos** mejora ligeramente con respecto al detector Diagonal, obtiene un 3,02% más de aciertos, y una mejoría más apreciable comparado con el detector Identidad, puesto que alcanza un 15,85% más de aciertos,. Por último, vemos como la clase **Monólogos** es la que consigue un resultado mucho mejor en comparación con los otros dos detectores, disminuyendo el porcentaje de error con respecto a la clase **Fútbol**. Numéricamente vemos como consigue un 16,02% más que el detector Diagonal y un 25,29 % en comparación al detector Identidad.

La Matriz de confusión 1.1.5.a muestra también las desviaciones típicas, las cuáles afirman que los resultados obtenidos con el detector Multidimensional obtienen los mejores resultados, puesto que a parte de conseguir porcentajes de acierto más elevados en media, las desviaciones son menores y están más cerca de los valores obtenidos en media.

| Audios/Test Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|------------------------------|---------------|------------|---------------|------------|
| Cuentos | 98 ± 4 | 0 | 0 | 2 ± 4 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 1 ± 3 | 0 | 94 ± 8 | 5 ± 9 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 1.1.6.a

En la Matriz de confusión 1.1.6.a se comprueba como por audios completos mejora notablemente la clasificación. Tanto en la clase **Noticias** como en **Monólogos** el 100% de los audios son clasificados correctamente, y en las clases **Cuentos** y **Fútbol** obtienen unos porcentajes elevados (98% y 94% respectivamente).

Comparando estos resultados a los conseguidos en la clasificación por trozos, mostrados en la matriz de confusión 1.1.5.a, la mejora es evidente. La clase **Cuentos** obtiene un 14,24% más de aciertos, la clase **Noticias** un 4,81%, **Fútbol** un 18,46% y **Monólogos** 8,07%.

Contrastando ahora estos datos con los obtenidos en la clasificación por audios de los dos detectores anteriores observamos como también por audios el detector Multidimensional obtiene mejores resultados. La clase **Fútbol** continúa teniendo el peor resultado, clasificándose algunos audios de dicha clase como **Monólogos**, pero ha mejorado notablemente con respecto a los porcentajes hallados anteriormente.

Conclusiones experimento 1:

Comparando los datos de los tres detectores, se ve que el detector Diagonal mejora los datos del detector Identidad, y que el detector Multidimensional mejora los porcentajes de los dos anteriores, por lo que se puede afirmar que el detector con el que se obtienen los mejores resultados es el Multidimensional. Este detector, tal y como se indicó anteriormente, es el que consideraba que los datos de entrenamiento provienen de una gaussiana multidimensional. Hemos podido comprobar que tanto en la clasificación por trozos como por audios es con este detector donde los porcentajes de acierto son superiores en todas las clases. Todo esto puede deberse a varios motivos:

- 1) Que con la covarianza diagonal se obtengan mejores resultados que con la identidad se puede deber a que cada variable tiene una varianza diferente y por esto influye de distinta manera en el clasificador, sin embargo para la covarianza identidad son los mismos valores y esto no sucede.
- 2) Que la covarianza multidimensional mejore con respecto a la diagonal nos indica que existen correlaciones entre las variables de entrada y que la covarianza multidimensional al ser una matriz “completa”, es decir, con datos en todas las filas y columnas, es capaz de capturar todas esas relaciones y explotarlas para mejorar los resultados.

Con los tres detectores la clase mejor clasificada es **Noticias** y la peor **Fútbol**.

4.4.1.2 Experimento 2

DETECTOR IDENTIDAD

| Trozos/Test Mitad/Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 82,14 | 2,47 | 10,44 | 4,95 |
| Noticias | 0,89 | 93,11 | 2,44 | 3,56 |
| Fútbol | 18,05 | 4,76 | 64,91 | 12,28 |
| Monólogos | 11,36 | 5,79 | 6,46 | 76,39 |

Matriz de confusión 1.2.1.a

Los porcentajes más elevados se encuentran en la diagonal principal. Comparando los resultados con el detector Identidad del primer apartado, los datos aumentan en los cuatro tipos de narraciones: la clase mejor clasificada es **Noticias**, que obtiene un 5,32% más de aciertos que en el apartado anterior, la clase **Cuentos** y **Fútbol** son las que consiguen mayores porcentajes de mejora, un 10,94 % y un 14,62% respectivamente, y **Monólogos** crece en un 3,43%.

La clase con más problemas para ser diferenciada sigue siendo **Fútbol** y, aunque ha mejorado, continúa confundándose con las clases **Cuentos** y **Monólogos**. Se puede ver que aunque los trozos clasificados como **Monólogos** han disminuido, los errores con respecto a la clase **Cuentos** han aumentado. Es curioso observar como en la fila de la clase **Monólogos** sucede algo parecido, disminuye el porcentaje de fallo clasificando estos trozos como **Fútbol** y sin embargo crece el porcentaje con respecto a **Cuentos**.

DETECTOR DIAGONAL

| Trozos/Test Mitad/Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|-------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 91,76 | 1,37 | 1,65 | 5,22 |
| Noticias | 0,67 | 98,44 | 0,22 | 0,67 |
| Fútbol | 7,27 | 1,25 | 81,96 | 9,52 |
| Monólogos | 2,45 | 4,23 | 7,35 | 85,97 |

Matriz de confusión 1.2.2.a

Tal y como pasó en el experimento anterior, los resultados mejoran con respecto al detector Identidad. Nuevamente la clase **Noticias** es la que logra un mejor resultado, consiguiendo un 5,72% de aciertos más que el detector Identidad. Las clases **Cuentos** y **Monólogos** también mejoran considerablemente, un 11,71% y un 12,54% respectivamente. La clase **Cuentos**, aunque continúa siendo la que obtiene un porcentaje de acierto más bajo, consigue un 26,26% más que el detector Identidad.

Observando ahora los errores cometidos, se puede ver como en la clase **Cuentos** se ven reducidos los trozos que son clasificados como **Fútbol** y **Noticias**, incrementándose ligeramente los **Monólogos**. En la clase **Fútbol** los errores siguen recayendo sobre las clases **Monólogos** y **Cuentos** mayoritariamente, aunque los porcentajes de error son menores. En la clase **Monólogos** el porcentaje más grande de errores cae sobre la clase **Fútbol**, en vez de sobre **Cuentos** como pasaba en el detector Identidad.

Comparando ahora los resultados con los del experimento anterior se puede ver como disminuyen los errores y mejoran los resultados de las cuatro clases: **Cuentos** un 10,20%, **Noticias** un 3,06%, **Fútbol** un 6,31% y **Monólogos** un 7,79%.

DETECTOR MULTIDIMENSIONAL

| Trozos/Test Mitad/Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|---------------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 90,11 | 4,12 | 1,92 | 3,85 |
| Noticias | 1,11 | 98,45 | 0,22 | 0,22 |
| Fútbol | 1,25 | 0,75 | 89,73 | 8,27 |
| Monólogos | 1,56 | 0 | 8,24 | 90,20 |

Matriz de confusión 1.2.3.a

Utilizando la covarianza Multidimensional mejoran los resultados con respecto a los dos detectores anteriores. La clase **Noticias** vuelve a ser la clase mejor clasificada y **Fútbol** la peor, pero esta vez con un porcentaje bastante elevado que casi consigue igualar a las clases **Cuentos** y **Monólogos**.

Comparando las soluciones con las del detector Identidad, los aciertos crecen considerablemente: la clase **Cuentos** mejora un 9,70%, **Noticias** un 5,73%, **Fútbol** un 38,23% y **Monólogos** un 18,07%. Si se comparan ahora con el detector Diagonal, se puede ver como la clase **Cuentos** obtiene un 1,79% menos de aciertos, y la clase **Noticias** únicamente un 0,01%, pero en cambio, **Fútbol**, que es la clase que más dificultades tiene para clasificarse, obtiene un 9,48% más de aciertos, y **Monólogos** un 4,92%.

Mirando ahora los valores fuera de la diagonal principal se puede ver como las clases **Fútbol** y **Monólogos** tienen un porcentaje de error similar entre sí, de un ~8,2%. En el resto de clases se verifican menos errores, a excepción de la clase **Cuentos**, donde sube el porcentaje de trozos clasificado como **Noticias**.

Realizando una comparativa de los resultados con los del experimento 1, se ve como mejoran los datos de todas las clases excepto el porcentaje de la clase **Monólogos**, que se ve reducido un 2,58%, puesto que aumentan los errores cometidos con respecto a la clase **Fútbol**.

Conclusiones experimento 2:

Considerando todo lo que se ha comentado anteriormente se puede afirmar que usar mitades de audios para entrenar y testear mejora respecto a usar audios completos sensiblemente, pero no tanto como se esperaba. Esto lejos de ser un mal resultado indica que la aproximación que se está realizando para clasificar género es buena, porque se está capturando información relevante para discriminar el género, independientemente de que utilice partes de cada audio para entrenar o no. Si no fuese así, metiendo trozos del mismo audio en los datos de entrenamiento y test tendría unas mejoras espectaculares porque sería capaz de separar las clases separando los audios individuales.

4.4.1.3 Experimento 3

Las matrices de confusión que pongo a continuación son la media y la desviación típica de esas 10 repeticiones:

DECISOR IDENTIDAD

| Trozos/test Aleatorio/Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|------------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 81,84 ± 1,62 | 2,42 ± 0,5 | 11,84 ± 1,1 | 3,9 ± 0,69 |
| Noticias | 1,27 ± 0,33 | 90,42 ± 0,84 | 1,89 ± 0,68 | 6,42 ± 0,87 |
| Fútbol | 13,86 ± 1,83 | 5,71 ± 0,85 | 65,79 ± 2,75 | 14,64 ± 1,63 |
| Monólogos | 10,58 ± 1,84 | 6,59 ± 0,78 | 8,44 ± 1,43 | 74,39 ± 1,54 |

Matriz de confusión 1.3.1.a

En la Matriz de confusión 1.3.1.a se puede ver como los datos mejoran considerablemente con respecto al primer experimento: un 10,53% para la clase **Cuentos**, un 2,28% para la clase **Noticias**, un 16,17% en la clase **Fútbol** y muy ligeramente en la clase **Monólogos** (un 0,73%). Sin embargo, comparando los resultados con los obtenidos en el segundo experimento, los datos son bastante similares: la clase **Cuentos**, la clase **Noticias** y la clase **Monólogos** obtienen un porcentaje de aciertos menor (un 0,36%, 2,88% y 2,61% respectivamente). La clase **Fútbol** es la única que tiene un resultado algo superior en media: un 1,35% más de aciertos.

Observando ahora los valores fuera de la diagonal principal, se puede ver como un 11,84% de trozos de la clase **Cuentos** son clasificados como **Fútbol**. Como se indicaba anteriormente, en la clase **Noticias** se obtiene un 2,88% menos de aciertos que en el experimento 2, y se puede ver como la bajada del porcentaje de aciertos se ha traducido en más errores con respecto a la clase **Monólogos**. La clase **Fútbol** sigue siendo la que comete más fallos, destacan sobretudo los trozos clasificados como **Cuentos** y **Monólogos** (13,86% y 14,64%). En la clase **Monólogos** los errores se reparten en las tres clases, siendo más numeroso el número de errores con respecto a la clase **Cuentos**.

A parte de que los porcentajes en media son bastante mejores que los obtenidos en el experimento 1, si se comparan las desviaciones típicas se puede ver como las halladas en este apartado están muy próximas a la media, lo cual es muy positivo para corroborar que utilizando datos de entrenamiento formado por trozos de audio cogidos de forma aleatoria hace que tengamos porcentajes de acierto más altos.

DECISOR DIAGONAL

| Trozos/test Aleatorio/Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 90,16 ± 1,04 | 2,31 ± 0,5 | 3,11 ± 0,39 | 4,42 ± 0,7 |
| Noticias | 0,85 ± 0,84 | 97,78 ± 0,57 | 0,53 ± 0,21 | 0,84 ± 0,33 |
| Fútbol | 4,11 ± 1,00 | 1,21 ± 0,74 | 85,73 ± 1,46 | 8,95 ± 1,23 |
| Monólogos | 1,02 ± 0,41 | 5,72 ± 0,76 | 10,94 ± 1,32 | 82,32 ± 1,51 |

Matriz de confusión 1.3.2.a

Se observa como mejoran los datos con respecto al decisor Identidad: un 10,16% en la clase **Cuentos**, 8,13% en **Noticias**, 30,30% en **Fútbol** y un 10,66% en **Monólogos**. También vemos como los errores se han visto reducidos considerablemente en todas las clases. Destaca la clase **Monólogos**, en la que en el decisor Identidad se veía un porcentaje de error de 10,58% con respecto a la clase **Cuentos**, y sin embargo en este decisor este porcentaje se ha visto reducido a un 1,02%, en cambio el porcentaje de errores con respecto a la clase **Fútbol** ha aumentado de un 8,44% a un 10,94%.

Comparando con el decisor Diagonal del primer experimento, la mejoría es también bastante notable en aciertos: un 8,28% en la clase **Cuentos**, 2,37% en **Noticias**, 11,20% en **Fútbol** y un 3,22% en **Monólogos**, lo que se traduce también en una disminución de los errores.

Mirando los datos del segundo experimento, sucede como en el decisor Identidad, los resultados son bastante similares. Tan solo mejora el porcentaje de aciertos para la clase **Fútbol** (un 4,59% más aciertos), lo que es bastante interesante ya que como se está viendo a lo largo de todo este capítulo, es la clase que más errores comete. Las otras tres clases están por debajo: **Cuentos** un 1,74% menos de aciertos, **Noticias** un 0,67% y **Monólogos** un 4,24%.

Se puede ver como los resultados obtenidos en media son bastante fiables puesto que las desviaciones típicas halladas son valores muy pequeños, es decir, todos los cálculos realizados se hallaban bastante próximos a la media.

DECISOR MULTIDIMENSIONAL

| Trozos/test Aleatorio/Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|--|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 95,16 ± 1,35 | 1,29 ± 0,49 | 1,13 ± 0,54 | 2,42 ± 1,18 |
| Noticias | 0,65 ± 0,51 | 99,02 ± 0,55 | 0,13 ± 0,21 | 0,2 ± 0,22 |
| Fútbol | 0,68 ± 0,24 | 0,12 ± 0,18 | 92,38 ± 1,70 | 6,82 ± 1,58 |
| Monólogos | 0,74 ± 0,30 | 0,18 ± 0,20 | 3,27 ± 0,95 | 95,81 ± 0,94 |

Matriz de confusión 1.3.3.a

Los datos mejoran visiblemente con respecto a los dos decisores anteriores, en concreto, con el decisor Diagonal: un 5,54% para la clase **Cuentos**, 1,26% en la clase **Noticias**, 7,75% para **Fútbol** y un 16,38% para la clase **Monólogos**. Los porcentajes de error también se han visto reducidos significativamente.

Comparando los resultados con los del decisor Multidimensional del primer experimento, vemos como los porcentajes de acierto son mejores en este tercer experimento. Los porcentajes de mejora son: un 10,93% para la clase **Cuentos**, 3,78% para la clase **Noticias**, 16,42% para **Fútbol** y por último, un 3,54% para la clase **Monólogos**. Los errores cometidos, sobretudo en la clase **Fútbol**, se han visto reducidos considerablemente: en el primer experimento un 17,29% de locuciones de **Fútbol** eran clasificadas como **Monólogos**, sin embargo en este experimento este porcentaje ha sido disminuido a un 6,82%.

Al contrario de lo que ocurría en los dos decisores anteriores, en los que los resultados eran bastante similares a los obtenidos en el segundo experimento y no se visualizaba mejoría alguna, con la covarianza Multidimensional si que obtienen porcentajes de acierto superiores: un 5,60% para la clase **Cuentos**, 0,57% para la clase **Noticias**, 2,95% para la clase **Fútbol** y un 6,21% para la clase **Monólogos**.

Observando las desviaciones se puede afirmar que en todas las repeticiones realizadas los datos obtenidos están muy próximos a la media.

Conclusiones experimento 3:

Se puede afirmar que cogiendo los trozos de lugares aleatorios los resultados de los decisores Identidad y Diagonal no cambian demasiado con respecto al segundo experimento pero si se ve una notable mejoría en el decisor Multidimensional. Esto es porque los audios cambian algo de la primera parte a la segunda y por eso si se cogen los trozos de forma aleatoria se está muestreando todo el audio y el clasificador es capaz de separar los audios individuales. Por lo que la mejora puede que no se deba a que se captura mejor las características que dependen del género, sino a que se es capaz de separar los audios individuales.

4.4.1.4 Experimento 4

4.3.1.4.1 Sin clase Cuentos

En primer lugar se van a poner todos los **Cuentos** en el test, junto con la mitad de los datos de **Noticias**, **Fútbol** y **Monólogos**, y como datos de entrenamiento únicamente la otra mitad de los datos de **Noticias**, **Fútbol** y **Monólogos**.

DECISOR IDENTIDAD

| Trozos/Test Identidad | Noticias | Fútbol | Monólogos |
|-----------------------|--------------|---------------|---------------|
| Cuentos | 7,85 ± 2,25 | 54,04 ± 12,23 | 38,11 ± 12,57 |
| Noticias | 89,05 ± 4,78 | 2,89 ± 2,01 | 8,06 ± 3,28 |
| Fútbol | 6,23 ± 4,24 | 68,44 ± 8,21 | 25,33 ± 11,51 |
| Monólogos | 6,91 ± 2,47 | 13,37 ± 3,81 | 79,72 ± 5,05 |

Matriz de confusión 1.4.1.a

Nos fijaremos principalmente en la fila de la clase **Cuentos**. Se puede ver como los **Cuentos** se clasifican mayoritariamente como **Fútbol** en un 54,04%, seguido de la clase **Monólogos** con un 38,11%. Pocos **Cuentos** se clasifican como **Noticias** (solo un 7,85%). Observando las desviaciones se confirma que los **Cuentos** se clasifican principalmente como **Fútbol** y **Monólogos** (54,04% ± 12,23% y 38,11% ± 12,57%).

Respecto al resto de clases, se observa como **Noticias** continúa siendo la clase mejor clasificada y **Fútbol** la peor (esta última confundiéndose en un 25,33% con la clase **Monólogos**, y la clase **Monólogos** a su vez un 13,37% con la clase **Fútbol**).

| Audios/Test Identidad | Noticias | Fútbol | Monólogos |
|-----------------------|----------|---------|-----------|
| Cuentos | 0 | 63 ± 19 | 37 ± 19 |
| Noticias | 100 | 0 | 0 |
| Fútbol | 5 ± 10 | 82 ± 13 | 13 ± 13 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.2.a

Por audios se puede ver como el 63% de los mismos son clasificados como **Fútbol** y el 37% son clasificados como **Monólogos**, por lo que al igual que en la clasificación por trozos, los **Cuentos** son clasificados como **Fútbol** principalmente. La desviación típica por audios también corrobora lo indicado en el comentario anterior.

DECISOR DIAGONAL

| Trozos/Test Diagonal | Noticias | Fútbol | Monólogos |
|----------------------|--------------|--------------|--------------|
| Cuentos | 14,20 ± 4,04 | 42,26 ± 8,67 | 43,54 ± 8,73 |
| Noticias | 96,15 ± 5,85 | 1,41 ± 2,52 | 2,44 ± 3,61 |
| Fútbol | 2,08 ± 1,21 | 83,56 ± 9,34 | 14,36 ± 9,30 |
| Monólogos | 6,18 ± 2,92 | 13,22 ± 5,91 | 80,6 ± 8,63 |

Matriz de confusión 1.4.3.a

En este decisor la clase **Cuentos** se reparte a partes iguales entre las clases **Fútbol** y **Monólogos** (42,26% y 43,54%), siendo la probabilidad de clasificar **Cuentos** como **Monólogos** ligeramente superior (~1,30 %). Observando las desviaciones típicas de estas dos clases se puede ver como son también bastante similares por lo que no cambia lo comentado anteriormente. Solo un 14,20% de los trozos de la clase **Cuentos** son clasificados como **Noticias**.

| Audios/Test Diagonal | Noticias | Fútbol | Monólogos |
|----------------------|----------|---------|-----------|
| Cuentos | 6 ± 9 | 46 ± 17 | 48 ± 18 |
| Noticias | 99 ± 3 | 0 | 1 ± 3 |
| Fútbol | 0 | 94 ± 10 | 6 ± 10 |
| Monólogos | 0 | 2 ± 6 | 98 ± 6 |

Matriz de confusión 1.4.4.a

Con respecto a la clasificación por audios, el resultado es similar al de por trozos, se reparte también entre **Fútbol** y **Monólogos** (46% y 48%), siendo ligeramente superior la clasificación como **Monólogos**. Teniendo en cuenta las desviaciones típicas se confirma que la clase **Cuentos** se clasifica principalmente por las dos clases indicadas.

DECISOR MULTIDIMENSIONAL

| Trozos/Test Multidimensional | Noticias | Fútbol | Monólogos |
|------------------------------|--------------|--------------|--------------|
| Cuentos | 10,85 ± 3,10 | 48,47 ± 8,72 | 40,68 ± 9,05 |
| Noticias | 95,49 ± 6,16 | 1,98 ± 2,88 | 2,53 ± 3,65 |
| Fútbol | 1,79 ± 0,98 | 83,94 ± 9 | 14,27 ± 9 |
| Monólogos | 6,04 ± 3,23 | 12,12 ± 5,23 | 81,84 ± 8,24 |

Matriz de confusión 1.4.5.a

Utilizando una gaussiana con matriz Multidimensional se puede observar como los trozos de la clase **Cuentos** son clasificados como **Fútbol** en un 48,47% de los casos, seguidos de **Monólogos** (40,68%) y por último de **Noticias** (10,85%). Teniendo en cuenta las desviaciones típicas, algunos de los valores de la clase **Fútbol** y **Monólogos** se ven solapados en el intervalo de resultados hallado ($48,47\% \pm 8,72\%$ $40,68\% \pm 9,05\%$), lo que confirma que la clase **Cuentos** se clasifica principalmente como **Fútbol** y **Monólogos**.

| Audios/Test Multidimensional | Noticias | Fútbol | Monólogos |
|------------------------------|-------------|-------------|-------------|
| Cuentos | 33 ± 26 | 36 ± 16 | 31 ± 21 |
| Noticias | 100 | 0 | 0 |
| Fútbol | 0 | 98 ± 4 | 2 ± 4 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.6.a

En la clasificación por audios se puede ver como, en media, los datos están repartidos de forma equitativa y las desviaciones típicas corroboran este dato. Al igual que por trozos, la clase con mayor porcentaje de audios de la clase **Cuentos** es la clase **Fútbol** (un 36%), seguido de la clase **Noticias** y **Monólogos** (33% y 31% de audios respectivamente).

Conclusiones sin incluir clase Cuentos:

Teniendo en cuenta los datos obtenidos con los tres decisores se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Cuentos**, los trozos y los audios de dicha clase son clasificados principalmente como **Fútbol** y **Monólogos**. En los ML Identidad y Multidimensional las **Noticias** se clasifican primordialmente como **Fútbol** y en el ML Diagonal como **Monólogos**.

4.3.1.4.2 Sin clase Noticias

En este apartado se incluyeron todos los datos de la clase **Noticias** en el test, junto con la mitad de los datos de **Cuentos**, **Fútbol** y **Monólogos**, y como datos de entrenamiento el resto de los datos de las clases **Cuentos**, **Fútbol** y **Monólogos**.

DECISOR IDENTIDAD

| Trozos/Test Identidad | Cuentos | Fútbol | Monólogos |
|-----------------------|------------------|-------------------|-------------------|
| Cuentos | $79,27 \pm 8,92$ | $14,74 \pm 10,28$ | $5,99 \pm 3,15$ |
| Noticias | $7,55 \pm 5,14$ | $32,70 \pm 22,16$ | $59,75 \pm 18,50$ |
| Fútbol | $17,59 \pm 7,43$ | $60,81 \pm 10,80$ | $21,60 \pm 6,33$ |
| Monólogos | $9,74 \pm 4,84$ | $19,90 \pm 4,98$ | $70,36 \pm 6,28$ |

Matriz de confusión 1.4.7.a

Observando la fila de **Noticias** se ve como estas son clasificadas en media mayoritariamente como **Monólogos** (59,75%) y en segundo lugar como **Fútbol** (32,7%). Hay que tener en cuenta las desviaciones típicas ya que nos dan dos números elevados de 22,16% para la clase **Fútbol** y 18,50% para la clase **Monólogos**, por lo que con estos datos se puede afirmar con más motivo que la clase **Noticias** se divide entre **Fútbol** y **Monólogos** y en un pequeño porcentaje en **Cuentos**.

| Audios/Test Identidad | Cuentos | Fútbol | Monólogos |
|--------------------------|----------------|----------------|----------------|
| Cuentos | 77 ± 25 | 18 ± 22 | 5 ± 16 |
| Noticias | 6 ± 16 | 24 ± 29 | 70 ± 32 |
| Fútbol | 8 ± 4 | 78 ± 11 | 14 ± 12 |
| Monólogos | 3 ± 5 | 1 ± 2 | 96 ± 6 |

Matriz de confusión 1.4.8.a

Por audios la clase **Noticias** es clasificada como **Monólogos** en el 70% de los audios clasificados, y el 24% de los mismos son clasificados como **Fútbol**. Visto así la clase **Noticias** se clasifica como **Monólogos** en la mayoría de los casos, pero no puede pasar desapercibida la desviación típica calculada, ya que para **Fútbol** es del 29% y para **Monólogos** del 32%, por lo que aunque la mayoría de **Noticias** se clasifican como **Monólogos**, hay un buen porcentaje que se clasifica como **Fútbol** a tener en cuenta.

DECISOR DIAGONAL

| Trozos/Test Diagonal | Cuentos | Fútbol | Monólogos |
|-------------------------|---------------------|----------------------|---------------------|
| Cuentos | 86,25 ± 9,26 | 7,15 ± 6,66 | 6,60 ± 3,71 |
| Noticias | 24,39 ± 12,32 | 60,06 ± 18,88 | 15,55 ± 10,69 |
| Fútbol | 7,74 ± 4,42 | 78,84 ± 9,41 | 13,42 ± 8,45 |
| Monólogos | 1,84 ± 1,15 | 16,08 ± 8,88 | 82,08 ± 9,60 |

Matriz de confusión 1.4.9.a

La clase **Noticias** es clasificada mayoritariamente como **Fútbol** (60,06%), seguido de **Cuentos** (24,39%) y **Monólogos** (15,55%). Observando las desviaciones, los datos de la clase **Noticias** seguirían clasificándose como la clase **Fútbol** en un porcentaje superior que las otras dos clases.

| Audios/Test Diagonal | Cuentos | Fútbol | Monólogos |
|-------------------------|----------------|----------------|---------------|
| Cuentos | 91 ± 17 | 7 ± 13 | 2 ± 4 |
| Noticias | 20 ± 15 | 69 ± 23 | 11 ± 15 |
| Fútbol | 6 ± 5 | 88 ± 11 | 6 ± 10 |
| Monólogos | 0 | 4 ± 7 | 96 ± 7 |

Matriz de confusión 1.4.10.a

En la clasificación por audios la distribución es la misma que en la clasificación por trozos. Los audios de **Noticias** son clasificados en primer lugar como **Fútbol** (69% de los audios), seguidos de **Cuentos** (20%) y por último de **Monólogos** (11%). Las desviaciones nuevamente nos ayudan a confirmar que efectivamente los audios de la clase **Noticias** se clasifican como **Fútbol** principalmente.

DECISOR MULTIDIMENSIONAL

| Trozos/Test Multidimensional | Cuentos | Fútbol | Monólogos |
|------------------------------|---------------------|----------------------|---------------------|
| Cuentos | 83,85 ± 9,29 | 9,67 ± 8,27 | 6,48 ± 2,95 |
| Noticias | 19,14 ± 10,92 | 62,15 ± 19,04 | 18,71 ± 11,65 |
| Fútbol | 9 ± 4,63 | 77,98 ± 9,74 | 13,02 ± 7,90 |
| Monólogos | 2,56 ± 1,28 | 14,46 ± 8,04 | 82,98 ± 8,79 |

Matriz de confusión 1.4.11.a

Evitando meter los datos de entrenamiento de la clase **Noticias** se ve como los datos de test de esta clase son clasificados principalmente como datos de la clase **Fútbol** (62,15%). El resto de trozos están repartidos entre las otras dos clases, **Cuentos** y **Monólogos** (19,14% y 18,71%). Viendo las desviaciones típicas se confirma la clasificación comentada anteriormente.

| Audios/Test Multidimensional | Cuentos | Fútbol | Monólogos |
|------------------------------|----------------|----------------|---------------|
| Cuentos | 85 ± 19 | 14 ± 19 | 1 ± 3 |
| Noticias | 8 ± 9 | 81 ± 14 | 11 ± 13 |
| Fútbol | 6 ± 5 | 90 ± 7 | 4 ± 5 |
| Monólogos | 0 | 2 ± 6 | 98 ± 6 |

Matriz de confusión 1.4.12.a

Se observa como los audios de la clase **Noticias** son clasificados principalmente como **Fútbol**, al igual que pasó en la clasificación Multidimensional por trozos, aunque el porcentaje de acierto conseguido en la clasificación por audios es superior (81% frente al 62,15% por trozos). El resto de audios se reparte entre **Cuentos** y **Monólogos** (8% y 11% de los audios respectivamente). Teniendo en cuenta las desviaciones típicas se puede afirmar que no varían las conclusiones sacadas en las líneas anteriores.

Conclusiones sin incluir clase Noticias:

Teniendo en cuenta los datos obtenidos con los tres decisores se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Noticias**, los trozos y los audios de dicha clase son clasificados principalmente como **Fútbol** y **Monólogos**. En el clasificador ML Identidad las **Noticias** se clasifican primordialmente como **Monólogos**, y en los clasificadores ML Diagonal y Multidimensional como **Fútbol**.

4.3.1.4.3 Sin clase Fútbol

En tercer lugar se van a incluir todos los datos de la clase **Fútbol** en el test, junto con la mitad de los datos de **Cuentos**, **Noticias** y **Monólogos**, y como datos de entrenamiento únicamente la otra mitad de los datos de Cuentos, Noticias y Monólogos.

DECISOR IDENTIDAD

| Trozos/Test Identidad | Cuentos | Noticias | Monólogos |
|-----------------------|---------------------|---------------------|---------------------|
| Cuentos | 86,41 ± 3,90 | 3,31 ± 1,76 | 10,28 ± 3,83 |
| Noticias | 1,46 ± 0,55 | 89,64 ± 3,69 | 8,90 ± 3,97 |
| Fútbol | 30,23 ± 5,34 | 13,32 ± 1,20 | 56,45 ± 5,00 |
| Monólogos | 11,47 ± 5,44 | 6,96 ± 1,71 | 81,57 ± 4,41 |

Matriz de confusión 1.4.13.a

Contemplando la fila de la clase **Fútbol** se puede ver como los trozos de esta clase, en media, son clasificados principalmente como **Monólogos** (56,45%), seguidos de **Cuentos** (30,23%) y **Noticias** (13,32%). En este caso observando las desviaciones la clasificación no cambia, es decir que en la mayoría de las repeticiones realizadas, los datos de la clase **Fútbol** son clasificados primordialmente como **Monólogos**. Utilizando solamente los datos de entrenamiento se puede ver como los resultados son bastante similares.

| Audios/Test Identidad | Cuentos | Noticias | Monólogos |
|-----------------------|---------------|------------|----------------|
| Cuentos | 97 ± 8 | 0 | 3 ± 8 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 13 ± 8 | 21 ± 5 | 66 ± 10 |
| Monólogos | 3 ± 5 | 1 ± 2 | 96 ± 6 |

Matriz de confusión 1.4.14.a

Por audios, la clase **Fútbol** es clasificada mayoritariamente como **Monólogos** (66% de los audios), seguido de **Noticias** (21%) y **Cuentos** (13%), por lo que coincide con la clasificación por trozos en que los datos de **Fútbol** son principalmente clasificados como **Monólogos**, pero sin embargo la segunda clase más clasificada es **Noticias** al contrario que por trozos que era la clase **Cuentos**. Las desviaciones típicas no aportan información adicional, **Fútbol** se sigue catalogando como **Monólogos** en la mayoría de los casos.

DECISOR DIAGONAL

| Trozos/Test Diagonal | Cuentos | Noticias | Monólogos |
|----------------------|---------------------|---------------------|----------------------|
| Cuentos | 89,01 ± 4,98 | 3,62 ± 2,06 | 7,37 ± 4,73 |
| Noticias | 1,53 ± 1,35 | 96,23 ± 4,80 | 2,24 ± 3,50 |
| Fútbol | 27,53 ± 8,98 | 16,36 ± 3,66 | 56,11 ± 10,37 |
| Monólogos | 4,17 ± 2,93 | 8,04 ± 5,19 | 87,79 ± 7,52 |

Matriz de confusión 1.4.15.a

Por trozos, la clase **Fútbol** es clasificada especialmente como **Monólogos** en un 56,11 %. El resto de trozos se divide entre la clase **Cuentos** principalmente (27,53%) y la clase Noticias (16,36%). Aunque las desviaciones típicas tanto de la clase **Cuentos** como de **Monólogos** son considerables, se seguirá afirmando que los datos de la clase **Fútbol** son clasificados, cuando no son incluidos sus datos en el entrenamiento, como **Monólogos**.

| Audios/Test/Media | Cuentos | Noticias | Monólogos |
|-------------------|-----------|-----------|----------------|
| Cuentos | 97 | 0 | 3 |
| Noticias | 0 | 99 | 1 |
| Fútbol | 16 ± 15 | 27 ± 8 | 57 ± 14 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.16.a

Los audios de la clase **Fútbol** son clasificados también primordialmente como **Monólogos** (57% de los audios), siendo un porcentaje similar al obtenido en la clasificación por trozos. En cambio, los porcentajes que se reparten entre las otras dos clases lo hacen de manera contraria a como lo hacían por trozos: el 16% de los audios son clasificados como **Cuentos** y 27% son clasificados como **Noticias**. Teniendo en cuenta las dispersiones de las clases **Cuentos** y **Monólogos** (16%±15% y 57%±14%) se ve como no hay valores que solapen, las locuciones de **Fútbol** son catalogadas como **Monólogos** en la mayoría de los casos.

DECISOR MULTIDIMENSIONAL

| Trozos/Test Multidimensional | Cuentos | Noticias | Monólogos |
|------------------------------|---------------------|---------------------|----------------------|
| Cuentos | 89,06 ± 4,48 | 3,53 ± 2,14 | 7,41 ± 3,94 |
| Noticias | 1,20 ± 0,60 | 96,01 ± 4,65 | 2,79 ± 4,29 |
| Fútbol | 31,35 ± 9,41 | 16,11 ± 3,66 | 52,54 ± 10,59 |
| Monólogos | 4,91 ± 2,92 | 7,66 ± 5,07 | 87,43 ± 7,41 |

Matriz de confusión 1.4.17.a

Utilizando el clasificador ML Multidimensional se puede ver como los trozos pertenecientes a los audios de la clase **Fútbol** son clasificados como **Monólogos** en un 52,54 %. El resto de trozos se reparte entre la clase **Cuentos** (31,35%) y la clase **Noticias** (16,11%). Observando las desviaciones típicas se puede afirmar que los resultados obtenidos en la clase **Cuentos** y **Monólogos** no llegan a solapar ($31,35\% \pm 9,41\%$ y $52,54\% \pm 10,59\%$), por lo que **Monólogos** es el que sigue recibiendo un mayor número de trozos de la clase **Cuentos** en la mayoría de las repeticiones realizadas.

| Audios/Test Multidimensional | Cuentos | Noticias | Monólogos |
|------------------------------|-------------------------------|-------------------------------|-------------------------------|
| Cuentos | 99 ± 3 | 0 | 1 ± 3 |
| Noticias | 0 | 99 ± 3 | 1 ± 1 |
| Fútbol | 20 ± 21 | 23 ± 13 | 57 ± 20 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.18.a

Los audios de la clase **Fútbol** se clasifican como **Monólogos** principalmente al igual que en la clasificación por trozos aunque en un porcentaje algo mayor (57%), siendo el resto de audios repartidos entre **Cuentos** y **Noticias** (20% y 23% de audios respectivamente). Se observa como las clases **Cuentos** y **Monólogos** tienen una dispersión parecida, por lo que nos fijaremos fundamentalmente en el valor obtenido en media, donde **Monólogos** obtiene el porcentaje más elevado de audios de **Fútbol**.

Conclusiones sin incluir clase Fútbol:

Teniendo en cuenta los datos obtenidos con los tres decisores se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Fútbol**, los trozos y los audios de dicha clase son clasificados principalmente como **Monólogos**. Es curioso observar como por trozos los datos de la clase **Fútbol** se clasifican, a parte de como **Monólogos**, como **Cuentos** en los tres decisores, y sin embargo, por audios, a parte de como **Monólogos** se suelen clasificar como **Noticias**.

4.3.1.4.4 Sin Monólogos

En último lugar se pondrán todos los datos de la clase **Monólogos** en el test, junto con la mitad de los datos de **Cuentos**, **Noticias** y **Fútbol**, y como datos de entrenamiento únicamente la otra mitad de los datos de **Cuentos**, **Noticias** y **Fútbol**. Este apartado será bastante interesante porque las otras tres clases eran clasificadas como **Monólogos** principalmente ante la ausencia de sus datos en el entrenamiento, en este caso se verá a que clase se parece más la clase **Monólogos**.

DECISOR IDENTIDAD

| Trozos/Test Identidad | Cuentos | Noticias | Fútbol |
|-----------------------|---------------------|---------------------|---------------------|
| Cuentos | 81,13 ± 9,78 | 3,06 ± 1,49 | 15,81 ± 10,80 |
| Noticias | 1,48 ± 0,54 | 93,26 ± 3,42 | 5,26 ± 3,22 |
| Fútbol | 22,57 ± 11,04 | 6,72 ± 4,44 | 70,71 ± 7,81 |
| Monólogos | 28,66 ± 6,78 | 14,79 ± 2,94 | 56,55 ± 4,48 |

Matriz de confusión 1.4.19.a

Se puede comprobar que al no introducir los datos de la clase **Monólogos** en los datos de entrenamiento, la mayoría de los trozos caen en la clase **Fútbol** (56,55%), seguidos de la clase **Cuentos** (28,66%) y de la clase **Noticias** (14,79%). Las desviaciones típicas no hacen que estos porcentajes varíen demasiado, por lo que se puede afirmar que, por trozos, la clase **Monólogos** se clasifica principalmente como **Fútbol**.

| Audios/Test Identidad | Cuentos | Noticias | Fútbol |
|-----------------------|----------------|------------|----------------|
| Cuentos | 83 ± 22 | 0 | 17 ± 22 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 15 ± 11 | 4 ± 6 | 81 ± 10 |
| Monólogos | 21 ± 12 | 4 ± 2 | 75 ± 11 |

Matriz de confusión 1.4.20.a

En la clasificación por audios el orden de clasificación es igual que por trozos: La clase **Monólogos** es clasificada como **Fútbol** en la mayor parte de los casos (75% de los audios), seguido de **Cuentos** (21%) y por último de **Noticias** (4%). En este caso, el porcentaje de clasificación por audios en **Fútbol** es más determinante que por trozos (de 56,55% a 75%). Las desviaciones típicas no varían las deducciones obtenidas.

DECISOR DIAGONAL

| Trozos/Test Diagonal | Cuentos | Noticias | Fútbol |
|----------------------|----------------------|---------------------|----------------------|
| Cuentos | 88,62 ± 8,13 | 3,54 ± 1,97 | 7,84 ± 8,38 |
| Noticias | 2,35 ± 2,18 | 95,89 ± 5,50 | 1,76 ± 3,55 |
| Fútbol | 10,15 ± 5,88 | 2,20 ± 1,15 | 87,65 ± 5,59 |
| Monólogos | 36,68 ± 14,69 | 6,75 ± 1,17 | 56,57 ± 13,95 |

Matriz de confusión 1.4.21.a

Sin incluir en el entrenamiento los datos de la clase **Monólogos** se puede comprobar como estos se clasifican en un 56,57% de los casos como si fueran de la clase **Fútbol** (porcentaje muy similar al obtenido en el decisor Identidad). Un 36,68% de datos son clasificados como **Cuentos**, y un pequeño porcentaje (6,75%) es clasificado como **Noticias**. Teniendo en cuenta las desviaciones típicas de los datos de la clase **Cuentos** y **Fútbol** se puede ver como en algunas repeticiones la clase **Monólogos** se reparte en mayor porcentaje en **Cuentos** que en **Fútbol**.

| Audios/Test Diagonal | Cuentos | Noticias | Fútbol |
|----------------------|----------------|------------|----------------|
| Cuentos | 92 ± 16 | 0 | 8 ± 16 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 6 ± 7 | 0 | 94 ± 7 |
| Monólogos | 33 ± 25 | 0 | 67 ± 25 |

Matriz de confusión 1.4.22.a

Clasificando los audios de la clase **Monólogos** se ve como estos se clasifican también principalmente como **Fútbol** (67%) al igual que por trozos, aunque con un porcentaje mayor, y como **Cuentos** en el resto de los casos (33%). La desviación típica es la misma tanto para **Cuentos** como para **Fútbol**, por lo que nos fijaremos únicamente en los datos obtenidos en media.

DECISOR MULTIDIMENSIONAL

| Trozos/Test Multidimensional | Cuentos | Noticias | Fútbol |
|------------------------------|----------------------|---------------------|---------------------|
| Cuentos | 86,86 ± 8,46 | 3,2 ± 1,90 | 9,94 ± 8,96 |
| Noticias | 2 ± 1,50 | 95,58 ± 5,45 | 2,42 ± 4,18 |
| Fútbol | 12,36 ± 6,83 | 1,97 ± 0,94 | 85,67 ± 6,57 |
| Monólogos | 42,87 ± 13,34 | 7,13 ± 1,20 | 50 ± 12,66 |

Matriz de confusión 1.4.23.a

Los trozos de la clase **Monólogos** son repartidos entre las tres clases restantes de la siguiente manera: un 50% a la clase **Fútbol**, un 42,87% y un 7,13% a la clase **Noticias**. Teniendo en cuenta las desviaciones se ve como aunque en media los trozos de **Monólogos** son clasificados como **Fútbol**, hay repeticiones en las que se clasifican mayoritariamente como **Cuentos**.

| Audios/Test Multidimensional | Cuentos | Noticias | Fútbol |
|------------------------------|----------------|------------|----------------|
| Cuentos | 95 ± 11 | 0 | 5 ± 11 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 7 ± 8 | 0 | 93 ± 8 |
| Monólogos | 52 ± 15 | 0 | 48 ± 15 |

Matriz de confusión 1.4.24.a

Los audios de **Monólogos** se reparten entre la clase **Cuentos** y la clase **Fútbol** (52% y 48% de los audios respectivamente). En la clasificación por trozos, el mayor porcentaje lo recibía la clase **Fútbol**, en cambio por audios, los **Monólogos** son clasificados como **Cuentos** principalmente, aunque los porcentajes estén bastante próximos entre sí. Si se comparan nuevamente estos resultados con los obtenidos en la clasificación por trozos, se ve como ningún audio es clasificado como **Noticias**. La desviación típica es la misma para la clase Cuentos como para **Fútbol**, por lo que nos seguiremos fijando básicamente en los resultados obtenidos en media.

Conclusiones sin incluir clase Monólogos:

Teniendo en cuenta los datos obtenidos con los tres decisores se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Monólogos**, los trozos y los audios de dicha clase son clasificados principalmente como **Fútbol**, a excepción del decisor ML Multidimensional, donde en la clasificación por audios es la clase **Cuentos** la que obtiene mayor porcentaje, siendo **Fútbol** la que consigue el siguiente porcentaje más alto.

4.4.2 Resultados clasificador Vecino más próximo

4.4.2.1 *K* óptima

La manera de hallar la *K* óptima es realizando un procedimiento de validación cruzada. En primer lugar, se ha elegido un conjunto de datos de entrenamiento formado por 40 audios, es decir, 10 audios de cada una de las cuatro clases (Cuentos, Noticias, Fútbol y Monólogos) y siendo este dividido en 5 subconjuntos que se llamarán *x1*, *x2*, *x3*, *x4* y *x5*. Se ha tenido especial cuidado en repartir equitativamente los datos de entrenamiento para que los cinco subconjuntos tengan datos de los cuatro estilos de narración. Se quiere elegir un valor del parámetro *K* entre los valores 1, 3, 5, 7, 9 y 11. Tal y como se comentó en el Capítulo 2 (apartado 2.4.2) es conveniente elegir valores de *K* impares para evitar empates a votos a la hora de contar los vecinos más cercanos.

Se harán 5 iteraciones de validación, en cada una se irá cambiando el conjunto de validación y el conjunto de entrenamiento. Para la primera iteración se ha cogido el subconjunto *x1*, el cuál va a hacer de conjunto de validación (se llamará *XV1*) y los subconjuntos de *x2* a *x5* que formarán el conjunto de entrenamiento (*XR*). Una vez decididos los conjuntos, lo que se tiene que hacer es calcular el error que se obtiene al clasificar los datos de *XV1* usando como entrenamiento los datos de *XR* y como parámetro cada una de las *K* que se indicaron anteriormente.

Para cada dato de *XV1* se tienen que coger los “*K*” datos de *XR* más cercanos a éste y ver cual es la etiqueta de la mayoría. Esta etiqueta es la que se asigna al dato de validación. Si coincide con la etiqueta del propio dato, se anota un acierto y si no se anota un fallo.

Seguidamente se realiza la segunda iteración de validación. Se coge el subconjunto *x2*, el cuál será ahora el conjunto de validación (*XV2*) y los subconjuntos *x1*, *x3*, *x4* y *x5* serán el nuevo *XR*. Se hace igual que con la primera iteración, se calculan los errores de validación para cada valor de *K* pero usando el nuevo conjunto de validación y de entrenamiento. En la tercera iteración se emplea el subconjunto *x3* como *XV3*, y el resto de los subconjuntos (*x1*, *x2*, *x4* y *x5*) como *XR*, la cuarta iteración se usa el *x4* como *XV4* y el resto (*x1*, *x2*, *x3* y *x5*) como *XR* y por último el *x5* como *XV5* y el resto de subconjuntos (*x1*, *x2*, *x3* y *x4*) como conjunto de datos de entrenamiento.

La matriz de confusión que se pone a continuación muestra los errores de cada conjunto de validación para cada parámetro K, y en la última fila la media de las 5 validaciones:

| | K=1 | K=3 | K=5 | K=7 | K=9 | K=11 |
|-------|---------------|--------------|--------------|---------------|---------------|---------------|
| XV1 | 37,29 | 39,83 | 43,78 | 43,22 | 44,63 | 44,07 |
| XV2 | 45 | 47 | 47 | 46 | 45 | 47 |
| XV3 | 36,16 | 33,33 | 33,61 | 32,48 | 33,05 | 34,46 |
| XV4 | 30,51 | 31,07 | 31,64 | 32,77 | 36,16 | 36,16 |
| XV5 | 37 | 35,87 | 36,72 | 36,16 | 39,59 | 38,98 |
| MEDIA | 37,192 | 37,42 | 38,55 | 38,126 | 39,686 | 40,134 |

El error más pequeño obtenido es para K=1, por lo que esta será la K óptima que se utilizará.

4.4.2.2 Experimento 1

| Trozos/Test | Cuentos | Noticias | Fútbol | Monólogos |
|-------------|--------------------|---------------------|---------------------|---------------------|
| Cuentos | 63,6 ± 4,14 | 1,38 ± 0,70 | 10,74 ± 4,22 | 24,28 ± 5,34 |
| Noticias | 5,88 ± 1,53 | 61,12 ± 7,40 | 9,64 ± 2,86 | 23,36 ± 8,21 |
| Fútbol | 11,43 ± 4,04 | 1,67 ± 0,72 | 50,29 ± 5,95 | 36,61 ± 5,35 |
| Monólogos | 8 ± 2,03 | 2,92 ± 0,98 | 14,4 ± 3,35 | 74,68 ± 4,62 |

Matriz de confusión 2.1.1.a

Se observa como la clase **Monólogos** es la que tiene un porcentaje superior de aciertos en media, un 74,68%, seguido de la clase **Cuentos** con un 63,6% de aciertos, **Noticias** con un 61,12% y por último **Fútbol** un 50,29%. Si bien los porcentajes más elevados se sitúan en la diagonal principal, llaman la atención los valores obtenidos fuera de ella. Resulta curioso como tanto en las clases **Cuentos**, **Noticias** y **Fútbol** existe un porcentaje elevado de trozos que recaen sobre la clase **Monólogos** (24,28%, 23,36% y 36,61%), que a su vez era la clase mejor clasificada. En el caso de la clase **Monólogos**, el porcentaje de error más numeroso recae sobre la clase **Fútbol**.

Las desviaciones típicas se aproximan bastante a la media. En la diagonal principal destacan sobretodo las obtenidas para la clase **Noticias** y **Fútbol**, siendo las más elevadas.

| Audios/Test | Cuentos | Noticias | Fútbol | Monólogos |
|-------------|---------------|----------------|----------------|------------|
| Cuentos | 97 ± 5 | 0 | 1 ± 2 | 2 ± 4 |
| Noticias | 0 | 93 ± 10 | 0 | 7 ± 10 |
| Fútbol | 7 ± 5 | 0 | 68 ± 12 | 25 ± 10 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 2.1.2.a

En la clasificación por audios se ve como mejoran los porcentajes de acierto para las cuatro clases. Observamos que la clase **Monólogos** continúa siendo la que tiene un porcentaje mayor, con la totalidad de los audios clasificados correctamente, seguida de la clase **Cuentos** (97% de aciertos), **Noticias** (93%) y **Fútbol** (68%). Los errores que se cometían al clasificar los trozos de las clases **Cuentos**, **Noticias** y **Fútbol** como **Monólogos**, se han visto reducidos al realizar la clasificación por audios enteros. Tan solo la clase **Fútbol** continúa teniendo un porcentaje a tener en cuenta, ya que el 25% de los audios de esta clase siguen siendo clasificados como **Monólogos**.

La mejora obtenida con respecto a la clasificación por trozos es muy llamativa: la clase **Cuentos** obtiene un 52,51% más de aciertos, similar a la mejora conseguida en la clase **Noticias** (un 52,15%). La clase **Fútbol** obtiene un 35,21% más de aciertos y la clase **Monólogos** un 33,90% consiguiendo clasificar los 10 audios de forma correcta.

Conclusiones experimento 1:

Con el clasificador KNN la clase con la que se obtienen mejores resultados, tanto en la clasificación por trozos como por audios es **Monólogos**, y la peor **Fútbol**.

4.4.2.3 Experimento 2

| Trozos/Test | Cuentos | Noticias | Fútbol | Monólogos |
|-------------|--------------|--------------|--------------|--------------|
| Cuentos | 71,43 | 1,37 | 6,32 | 20,88 |
| Noticias | 5,33 | 63,56 | 9,78 | 21,33 |
| Fútbol | 11,53 | 1,75 | 57,65 | 29,07 |
| Monólogos | 10,47 | 4,23 | 13,59 | 71,71 |

Matriz de confusión 2.2.1.a

Se observa como las clases **Monólogos** y **Cuentos** son las que obtienen un porcentaje de aciertos mayor, 71,71% y 71,43% respectivamente, seguidas de la clase **Noticias** con un 63,56%, y por último **Fútbol** un 57,65%.

Los porcentajes más elevados vuelven a situarse en la diagonal principal. Se puede ver como, al igual que paso en el Experimento 1, en las clases **Cuentos**, **Noticias** y **Fútbol** existe un porcentaje elevado de trozos clasificados erróneamente como **Monólogos** (20,88%, 21,33% y 29,07%). En cambio, los errores cometidos a la hora de clasificar la clase **Monólogos** están bastante repartidos entre las otras tres clases, es clasificada mayoritariamente como **Fútbol** y **Cuentos** (13,59% y 10,47%).

Comparando estas soluciones con los del Experimento 1 se ve como mejoran los aciertos de las clases **Cuentos**, **Noticias** y **Fútbol** (un 12,31%, 3,99% y 14,63% respectivamente). La clase **Monólogos** consigue un porcentaje de aciertos algo menor (un 3,97% menos), puesto que más trozos son clasificados como **Cuentos** y **Noticias**.

Conclusiones experimento 2:

Con el clasificador KNN la clase con la que se obtienen mejores resultados, tanto en la clasificación por trozos como por audios es **Monólogos**, y la peor **Fútbol**.

4.4.2.4 Experimento 3

| Trozos/Test Aleatorio | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 83,95 ± 2,33 | 0,36 ± 0,19 | 4,29 ± 0,74 | 11,40 ± 2,50 |
| Noticias | 3,98 ± 0,92 | 78,92 ± 1,79 | 5,49 ± 0,66 | 11,61 ± 1,40 |
| Fútbol | 3,61 ± 1,20 | 0,55 ± 0,35 | 80,44 ± 2,49 | 15,4 ± 2,07 |
| Monólogos | 4,23 ± 1,06 | 1,47 ± 0,64 | 7,39 ± 1,79 | 86,91 ± 1,50 |

Matriz de confusión 2.3.1.a

En este apartado se puede ver como mejoran notablemente los porcentajes de las cuatro clases con respecto a los dos apartados anteriores. Con respecto al primer apartado hay una mejora del 31,99% para la clase **Cuentos**, 29,12% para **Noticias**, 59,95% para **Fútbol** y 16,37% para **Monólogos**, y con respecto al segundo apartado la mejoría es de 17,52% para **Cuentos**, 24,16% para **Noticias**, 39,53% para **Fútbol** y 21,19% para **Monólogos**.

La clase mejor clasificada nuevamente es **Monólogos** (86,91%), seguida de **Cuentos** (83,95%), **Fútbol** (80,44%) y **Noticias** (78,92%). Destaca como en este caso, la clase **Fútbol** deja de ser la peor clasificada, obteniendo un aumento de aciertos considerable comparado con los dos apartados anteriores donde el porcentaje de aciertos apenas superaba la mitad de los trozos. Se puede observar también como ha descendido el porcentaje de errores de los trozos de las clases **Cuentos**, **Noticias** y **Fútbol** que eran clasificados como **Monólogos**. También disminuyen los errores en la clase **Monólogos**, el porcentaje de error con respecto a la clase **Fútbol** pasa de 14,4% y 13,59% a 7,39%.

Observando las desviaciones típicas se ve que los valores son bastante pequeños, lo que significa que en todos los cálculos realizados los resultados estaban bastante próximos a la media mostrada en la Matriz de confusión 2.3.1.a.

Conclusiones experimento 3:

Utilizando solamente los datos de entrenamiento, la totalidad de los trozos de las cuatro clases son clasificados correctamente, por lo que la dispersión es nula.

4.4.2.5 Experimento 4**4.4.2.5.1 Sin clase Cuentos**

| Trozos/Test | Noticias | Fútbol | Monólogos |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 3,47 ± 1,41 | 29,65 ± 7,41 | 66,88 ± 7,79 |
| Noticias | 63,91 ± 7,81 | 10,6 ± 3,21 | 25,49 ± 8,38 |
| Fútbol | 1,85 ± 0,85 | 55,12 ± 6,05 | 43,03 ± 6,35 |
| Monólogos | 3,07 ± 1,09 | 16,28 ± 4,10 | 80,65 ± 3,98 |

Matriz de confusión 2.4.1.a

Se observa que la clase **Cuentos** se clasifica mayoritariamente como **Monólogos** (66,88%), seguido de la clase **Fútbol** (29,65%) y por último de **Noticias** (3,47%). Observando las desviaciones típicas se puede comprobar como los valores no cambian en exceso, puesto que tanto la clase **Monólogos** como **Fútbol** tienen una desviación similar.

| Audios/Test | Noticias | Fútbol | Monólogos |
|-------------|----------------|----------------|----------------|
| Cuentos | 1 ± 32 | 18 ± 15 | 72 ± 28 |
| Noticias | 78 ± 31 | 1 ± 32 | 12 ± 15 |
| Fútbol | 1 ± 3 | 62 ± 26 | 37 ± 23 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.2.a

Con respecto a la clasificación por audios se puede observar como la clase **Cuentos** se clasifica principalmente como **Monólogos**, al igual que en la clasificación por trozos, aunque con un porcentaje mayor (un 7,15% más que por trozos). Teniendo en cuenta las desviaciones típicas llama la atención sobretudo la obtenida en la clase **Noticias**, ya que es un valor bastante elevado y distante del conseguido en media. No obstante, se confirma que los audios de la clase **Cuentos**, al no ser incluidos en los datos de entrenamiento, son clasificados como **Monólogos** en un porcentaje elevado de los casos.

Conclusiones sin incluir clase Cuentos:

Tanto en la clasificación por trozos como por audios completos, los datos de la clase **Cuentos** son clasificados como **Monólogos** ante la ausencia de estos dentro de los datos de entrenamiento.

4.4.2.5.2 Sin clase Noticias

| Trozos/Test | Cuentos | Fútbol | Monólogos |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 64,34 ± 4,03 | 10,91 ± 4,34 | 24,75 ± 5,51 |
| Noticias | 17,18 ± 3,78 | 25,49 ± 5,00 | 57,33 ± 5,09 |
| Fútbol | 11,50 ± 4,09 | 51,16 ± 6,16 | 37,34 ± 5,29 |
| Monólogos | 8,16 ± 2,01 | 14,82 ± 3,41 | 77,02 ± 4,89 |

Matriz de confusión 2.4.3.a

La mayor parte de los trozos de la clase **Noticias** son clasificados como **Monólogos** (57,33%), seguido de la clase **Fútbol** (25,49%) y por último de la clase **Cuentos** (17,18%). Las medidas de dispersión hacen que varíen ligeramente los resultados haciendo que la repartición entre **Cuentos** y **Fútbol** solape en algunas ocasiones ($17,18+3,78=20,96\%$ y $25,49-5=20,49\%$). Estos datos no influyen con respecto a la clase **Monólogos**, que es la que se lleva la mayoría de los votos tanto en media como observando las dispersiones.

| Audios/Test | Cuentos | Fútbol | Monólogos |
|-------------|---------------|----------------|---------------|
| Cuentos | 98 ± 5 | 1 ± 3 | 1 ± 3 |
| Noticias | 11 ± 16 | 10 ± 15 | 79 ± 8 |
| Fútbol | 6 ± 5 | 69 ± 16 | 25 ± 14 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.4.a

Al igual que en la clasificación por trozos, los audios de la clase **Noticias** se clasifican mayoritariamente como **Monólogos**, aunque en un porcentaje superior (79% frente a 57,33% por trozos). Si bien las desviaciones típicas de las clases **Cuentos** y **Fútbol** son mayores que la de **Monólogos**, esto no hace sino confirmar que las **Noticias** son clasificadas como **Monólogos** en la gran mayoría de los casos, ya que todos los resultados de las repeticiones realizadas del experimento están próximas a la media.

Conclusiones sin incluir clase Noticias:

Los datos de la clase **Noticias**, al no ser incluidos en el entrenamiento, se clasifican como **Monólogos** esencialmente, tanto por trozos como por audios completos

4.4.2.5.3 Sin clase Fútbol

| Trozos/Test | Cuentos | Noticias | Monólogos |
|-------------|--------------------|---------------------|---------------------|
| Cuentos | 70,64 ± 4,7 | 1,53 ± 0,68 | 27,83 ± 4,51 |
| Noticias | 6,68 ± 1,59 | 66,39 ± 7,82 | 26,93 ± 8,23 |
| Fútbol | 21,51 ± 2,50 | 3,84 ± 0,93 | 74,65 ± 2,64 |
| Monólogos | 10,22 ± 2,87 | 3,41 ± 1,11 | 86,37 ± 2,93 |

Matriz de confusión 2.4.5.a

La mayoría de los trozos de la clase **Fútbol** se clasifican como **Monólogos** (74,65%). Existen un porcentaje que se clasifica como **Cuentos** (21,51%) y una minoría que se clasifica como **Noticias** (3,84%). Se observa que las desviaciones típicas son pequeñas lo que significa que los resultados de las repeticiones realizadas están próximos a la media, por lo que se confirma que los trozos de la clase **Fútbol**, cuando no se incluyen sus datos en el entrenamiento, se clasifican como **Monólogos**. Se puede ver como los datos con respecto a la clase **Fútbol** no varían utilizando únicamente los datos de entrenamiento.

| Audios/Test | Cuentos | Noticias | Monólogos |
|-------------|---------------|----------------|------------|
| Cuentos | 96 ± 7 | 0 | 4 ± 7 |
| Noticias | 0 | 89 ± 14 | 11 ± 14 |
| Fútbol | 0 | 0 | 100 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.6.a

En el caso de la clasificación por audios se ve como la totalidad de los audios de **Fútbol** son clasificados como **Monólogos**, mejorando el porcentaje hallado en la clasificación por trozos, donde parte de estos trozos se clasificaban como **Cuentos**. Utilizando solo los datos de entrenamiento también se clasifican todas locuciones de **Fútbol** como **Monólogos**.

Conclusiones sin incluir clase Fútbol:

Al igual que sucedió en los dos apartados anteriores, al no incluir los datos de la clase **Fútbol** dentro del entrenamiento, estos son clasificados como **Monólogos** en la clasificación por trozos y por audios.

4.4.2.5.4 Sin clase Monólogos

| Trozos/Test | Cuentos | Noticias | Fútbol |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 82,31 ± 5,64 | 2,14 ± 0,91 | 15,55 ± 5,47 |
| Noticias | 8,89 ± 1,89 | 75,92 ± 4,76 | 15,19 ± 3,26 |
| Fútbol | 19,74 ± 7,24 | 3,73 ± 1,63 | 76,53 ± 6,86 |
| Monólogos | 32,13 ± 6,73 | 12,12 ± 1,91 | 55,75 ± 5,75 |

Matriz de confusión 2.4.7.a

Se puede ver como la clase **Monólogos** se clasifica en un porcentaje del 55,75% como **Fútbol**, un 32,13% se clasifica como **Cuentos** y 12,12% como **Noticias**. Teniendo en cuenta la desviación se observa como los trozos de la clase **Monólogos** se siguen clasificando mayoritariamente como **Fútbol**.

| Audios/Test | Cuentos | Noticias | Fútbol |
|-------------|----------------|------------|----------------|
| Cuentos | 96 ± 7 | 1 ± 3 | 3 ± 7 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 7 ± 7 | 0 | 93 ± 7 |
| Monólogos | 11 ± 15 | 0 | 89 ± 15 |

Matriz de confusión 2.4.8.a

Por audios los **Monólogos** también se clasifican como **Fútbol**, aunque en un porcentaje superior al que lo hacía por trozos. En esta ocasión 89% de los audios son clasificados como **Fútbol**, clasificándose un pequeño porcentaje como **Cuentos** (11%). La desviación típica es la misma para la clase **Cuentos** como para **Fútbol**, pero se puede ver como no varían en absoluto los resultados comentados anteriormente.

Conclusiones sin incluir clase Monólogos:

Los datos de la clase Monólogos son clasificados como Fútbol ante la ausencia de estos dentro de los datos de entrenamiento, tanto en la clasificación por trozos como por audios (en un porcentaje aún mayor)

4.4.3 Resultados clasificador Máquina de Vectores de Soporte

4.4.3.1 *C y gamma óptimas*

Para el Kernel Lineal es necesario hallar la C óptima, y para el Kernel Gaussiano se necesita hallar la C y la gamma óptimas. La forma de hallar estos parámetros es mediante la técnica de validación cruzada, al igual que hicimos en el clasificador KNN para hallar la K óptima. Utilizaremos el mismo conjunto de datos de entrenamiento utilizados en el KNN para poder comparar así los resultados entre ambos: conjunto formado por 10 audios de cada una de las clases, Cuentos, Noticias, Fútbol y Monólogos, que hacen un total de 40 audios. Este conjunto lo he dividido en 5 subconjuntos x1, x2, x3, x4 y x5, teniendo así cada conjunto datos de cada una de las clases. Haremos las 5 iteraciones de validación utilizando cada vez un subconjunto como conjunto de validación (XV), y los otros cuatro subconjuntos como conjunto de entrenamiento (XR) (ver validación cruzada del KNN).

Kernel Lineal:

Se quiere elegir un valor del parámetro C óptimo de entre los siguientes: 0.01 0.1 1 10 100 1000. Para ello hemos obtenido la siguiente matriz:

| | | | | | |
|--------|--------|--------|--------|--------|--------|
| 0.2712 | 0.1977 | 0.1808 | 0.1780 | 0.1893 | 0.2006 |
| 0.2910 | 0.2316 | 0.2175 | 0.2147 | 0.2260 | 0.2316 |
| 0.1695 | 0.1045 | 0.0904 | 0.1441 | 0.1780 | 0.1864 |
| 0.1638 | 0.1130 | 0.1215 | 0.1412 | 0.1497 | 0.1441 |
| 0.2288 | 0.1808 | 0.1864 | 0.2034 | 0.2175 | 0.2175 |

Cada fila representa los valores obtenidos utilizando uno de los subconjuntos como conjunto de validación y el resto como conjunto de entrenamiento. Las columnas representan los resultados obtenidos para cada valor de C propuesto. Hallando la media se obtiene:

0.2249 0.1655 **0.1593** 0.1763 0.1921 0.1960

Según este resultado se puede ver que el error más pequeño se obtiene con el valor de $C = 1$.

Kernel Gaussiano:

Ahora queremos obtener también un valor del parámetro C óptimo de entre los que indicamos en el Kernel Lineal: 0.01 0.1 1 10 100 1000, y de gamma óptima de entre: $(1/16*\sqrt{135})$ $(1/8*\sqrt{135})$ $(1/4*\sqrt{135})$ $(1/2*\sqrt{135})$ $(1*\sqrt{135})$ (siendo 135 el número de dimensiones de los datos). Para ello he obtenido la siguiente matriz:

3.3502 3.6129 3.7146 3.7458 3.7458
 1.5480 3.6129 3.7146 3.7458 3.7458
0.7965 1.3842 2.5481 3.7176 3.7458
 0.8051 1.3503 2.4237 3.7148 3.7402
 0.8051 1.3503 2.4237 3.7148 3.7402
 0.8051 1.3503 2.4237 3.7148 3.7402

Donde las filas representan los resultados obtenidos por los distintos valores de C, y las columnas los resultados por los distintos valores de gamma, de donde deducimos que los mejores valores de ambos parámetros son: $C=1$ y $\gamma = 1/16*\sqrt{135}$.

4.4.3.2 Experimento 1

KERNEL LINEAL

| Trozos/Test Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 82,19 ± 5,68 | 1,15 ± 0,72 | 10,29 ± 7,18 | 6,37 ± 2,65 |
| Noticias | 2,17 ± 1,14 | 91,39 ± 4,16 | 1,48 ± 0,66 | 4,96 ± 4,55 |
| Fútbol | 12,67 ± 5,72 | 2,01 ± 0,82 | 71,52 ± 8,56 | 13,8 ± 5,07 |
| Monólogos | 6,48 ± 2,45 | 3,30 ± 1,93 | 6,70 ± 1,72 | 83,52 ± 3,80 |

Matriz de confusión 3.1.1.a

En la Matriz de confusión utilizando el Kernel Lineal se ve como una vez más los valores más elevados se encuentran en la diagonal principal, por lo que se puede afirmar que por trozos, las cuatro clases logran clasificarse de forma correcta. **Noticias** es la clase con más porcentaje de aciertos (91,39 %), seguida de la clase **Monólogos** (83,52 %), **Cuentos** (82,19 %) y **Fútbol** (71,52 %).

Como curiosidades destacables se puede comprobar como un 10,29 % de los trozos de **Cuentos** son clasificados como **Fútbol**, y viceversa, un 12,67% de los trozos de Fútbol son clasificados como **Cuentos**. También se observa como algunos de estos trozos de la clase **Fútbol** son clasificados como **Monólogos** (un 13,8%).

A priori observando las desviaciones no aparecen grandes cambios. La desviación más alta se puede ver en la diagonal principal en la clase **Fútbol**, también se observa una desviación elevada en los fallos cometidos al clasificar la clase **Cuentos** como **Fútbol**.

| Audios/Test Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------|----------------|---------------|----------------|------------|
| Cuentos | 90 ± 14 | 0 | 10 ± 14 | 0 |
| Noticias | 0 | 99 ± 2 | 0 | 1 ± 2 |
| Fútbol | 8 ± 7 | 0 | 84 ± 12 | 8 ± 9 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 3.1.2.a

En la clasificación por audios los resultados mejoran con respecto a la clasificación por trozos. En este caso la clase mejor clasificada es la clase **Monólogos**, con la totalidad de los audios clasificados correctamente, un 19,73% más que por trozos. Le siguen las clases **Noticias** (99%) y **Cuentos** (90%). La clase **Fútbol**, aunque mejora en porcentaje con respecto a la clasificación por trozos, sigue siendo la clase peor clasificada con un 84% repartándose los errores entre las clases **Cuentos** y **Monólogos**.

Observando las desviaciones típicas destacan los valores hallados en la clase **Cuentos**, ya que aparece una dispersión del 14% tanto en aciertos como en errores al clasificar los **Cuentos** como locuciones de **Fútbol**. También en la clase **Fútbol** aparece una desviación del 12% con respecto a los aciertos, y otras de 7% y 9% con respecto a los errores cometidos con las clases **Cuentos** y **Monólogos**.

KERNEL GAUSSIANO

| Trozos/Test Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 70,38 ± 4,75 | 1,82 ± 1,03 | 12,23 ± 10,36 | 15,57 ± 9,55 |
| Noticias | 1,53 ± 1,01 | 87,71 ± 6,29 | 0,71 ± 0,37 | 10,05 ± 6,51 |
| Fútbol | 11,76 ± 4,58 | 2,13 ± 1,61 | 54,25 ± 8,57 | 31,86 ± 8,34 |
| Monólogos | 5,20 ± 3,22 | 3,48 ± 1,54 | 3,51 ± 2,41 | 87,81 ± 4,76 |

Matriz de confusión 3.1.3.a

Utilizando el Kernel Gaussiano se puede ver como las clases mejor clasificadas son **Monólogos** (87,81%) y **Noticias** (87,71%) con porcentajes bastante similares, seguidas de la clase **Cuentos** (70,38%) y de la clase **Fútbol** (54,25%).

Llama la atención el porcentaje elevado de trozos de la clase **Fútbol** que recaen sobre la clase **Monólogos** (31,86%) y la clase **Cuentos** (11,76%). Con respecto a la clase **Cuentos** a su vez se ve como se reparten trozos entre las clases **Fútbol** y **Monólogos** (12,23% y 15,57 %) respectivamente. En la clase **Noticias** existe un porcentaje a destacar clasificado como **Monólogos** (10,05 %). En la clase **Monólogos** se observa como los fallos están repartidos equitativamente entre las tres clases, recayendo algún trozo más en la clase **Cuentos**.

En la fila de la clase **Cuentos** aparecen valores de dispersión destacables con respecto a los errores cometidos al clasificar los **Cuentos** como **Fútbol** y **Monólogos**. Tanto en la clase **Noticias** como **Fútbol**, las desviaciones a tener en cuenta se encuentran en el porcentaje de aciertos y en el porcentaje de fallos con respecto a la clase **Monólogos**.

| Audios/Test Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------|----------------|------------|----------------|------------|
| Cuentos | 76 ± 22 | 0 | 21 ± 24 | 3 ± 7 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 7 ± 5 | 0 | 69 ± 15 | 24 ± 14 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 3.1.4.a

Con respecto a la clasificación por Audios podemos ver como al igual que en la clasificación por Trozos, las clases mejor clasificadas son **Noticias** y **Monólogos**, con la totalidad de los audios clasificados acertadamente. La clase **Cuentos** consigue clasificar un 76% de los audios de forma correcta, y por último, **Fútbol**, que es la clase peor clasificada, aunque con un porcentaje 13,88% mayor que en la clasificación por trozos.

Cabe destacar las dispersiones que aparecen en la fila de la clase **Cuentos**, un 22% de cara a los aciertos y un 24% por los fallos cometidos al clasificar los **Cuentos** como locuciones de **Fútbol**. Precisamente en la fila de la clase **Fútbol** aparecen también dos desviaciones típicas con un porcentaje considerable: un 15% en los aciertos y un 14% en los errores al clasificar los audios de **Fútbol** como **Monólogos**.

Conclusiones experimento 1:

En conjunto se puede afirmar que se obtienen mejores resultados utilizando el Kernel Lineal, ya que el porcentaje de aciertos para las cuatro clases es superior tanto en la clasificación por trozos como por audios.

Se puede ver como para ambos decisores las clases mejor clasificadas son **Noticias** y **Monólogos**: para la clasificación por trozos con el Kernel Lineal, **Noticias**, y por audios, **Monólogos**. En cambio, para el Kernel Gaussiano, para la clasificación por trozos la clase mejor clasificada es **Monólogos** y en la clasificación por audios hay empate entre las dos clases. La clase peor clasificada es de **Fútbol**.

4.4.3.3 Experimento 2

KERNEL LINEAL

| Trozos/Test Mitad/Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------------|--------------|--------------|--------------|-------------|
| Cuentos | 85,71 | 0,82 | 9,62 | 3,85 |
| Noticias | 2,22 | 95,56 | 0,44 | 1,78 |
| Fútbol | 9,27 | 1,75 | 81,96 | 7,02 |
| Monólogos | 8,24 | 1,78 | 6,68 | 83,3 |

Matriz de confusión 3.2.1.a

En este apartado podemos ver como los resultados mejoran comparándolos con los del apartado anterior. La clase mejor clasificada vuelve a ser **Noticias** que mejora un 4,56% y la peor, la clase **Fútbol**, aunque con un porcentaje de acierto más elevado (consigue un 14,59% más de aciertos). Entre ambas clases quedarían las clases **Cuentos** y **Monólogos**, la primera consigue un 4,28% más de aciertos, en cambio la segunda desciende ligeramente, obtiene un 0,26% menos.

Como resultados a destacar podemos ver como la mayoría de los errores en la clase **Cuentos** recaen sobre **Fútbol** (9,62%), y viceversa, los trozos de la clase **Fútbol** son clasificados como Cuentos (un 9,27 %) y como **Monólogos** (7,02%). Los errores de la clase **Monólogos** caen sobre las clases **Cuentos** y **Fútbol** (8,24% y 6,68% respectivamente).

KERNEL GAUSSIANO

| Trozos/Test Mitad/Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 78,02 | 1,1 | 9,89 | 10,99 |
| Noticias | 1,55 | 91,78 | 0,89 | 5,78 |
| Fútbol | 12,78 | 1,76 | 63,91 | 21,55 |
| Monólogos | 6,68 | 2,67 | 1,56 | 89,09 |

Matriz de confusión 3.2.2.a

En la matriz de confusión utilizando un Kernel Gaussiano observamos como la clase **Noticias** es la que obtiene el mayor porcentaje de aciertos (91,78%), seguido de la clase **Monólogos** (89,09%), **Cuentos** (78,02%) y por último de la clase **Fútbol** (63,91%). Comparando los resultados con los obtenidos en el apartado 1 podemos ver como ascienden todos los porcentajes de aciertos: en la clase **Cuentos** un 10,85%, **Noticias** un 4,64%, **Fútbol** un 17,80% y **Monólogos** un 1,45%.

Podemos ver como se reparten, aproximadamente a partes iguales, trozos de la clase **Cuentos** en las clases de **Fútbol** y **Monólogos**. Respecto a la clase **Noticias**, vemos como un 5,78% de los trozos son clasificados como **Monólogos**. Destacan los errores a la hora de clasificar los trozos de la clase **Fútbol**, puesto que un 21,55 % de los mismos se clasifican como **Monólogos** y un 12,78% como **Cuentos**. Y por último, los trozos de la clase **Monólogos** son confundidos principalmente por la clase **Cuentos** (6,68%).

Realizando una comparativa de estos datos con los resultados hallados con el Kernel Lineal, vemos como únicamente mejoran los aciertos de la clase **Monólogos** un 6,49%. El resto de clases consigue peores resultados: la clase **Cuentos** un 9,85% menos de aciertos, **Noticias** un 4,11% y **Fútbol** un 28,24%.

Conclusiones experimento 2:

Como pasó en el apartado anterior, se puede observar como tanto para el Kernel Lineal como para el Kernel Gaussiano la clase mejor clasificada es **Noticias**, y la clase peor clasificada es **Fútbol**. Aunque el porcentaje de la clase **Monólogos** es superior en el Kernel Gaussiano, se puede decir que el Kernel Lineal obtiene mejores resultados en las cuatro clases. Destaca sobretodo como mejora el resultado de la clase **Fútbol** ya que con el Kernel Lineal se obtiene un 28% de aciertos más que con el Kernel Gaussiano.

Comparando los datos con los obtenidos en el Experimento 1 se puede afirmar que utilizando la primera mitad del audio para entrenar y la otra mitad para testear se obtienen mejores resultados tanto en el Kernel Lineal como en el Gaussiano.

4.4.3.4 Experimento 3

KERNEL LINEAL

| Trozos/Test/ Aleatorio/Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 87,94 ± 1,28 | 0,68 ± 0,54 | 7,12 ± 0,90 | 4,26 ± 1,29 |
| Noticias | 1,65 ± 0,67 | 94,93 ± 1,10 | 1,11 ± 0,21 | 2,31 ± 0,81 |
| Fútbol | 7,55 ± 1,51 | 1,13 ± 0,57 | 82,92 ± 1,92 | 8,40 ± 1,69 |
| Monólogos | 5,68 ± 0,95 | 2,47 ± 0,56 | 6,02 ± 1,06 | 85,83 ± 1,57 |

Matriz de confusión 3.3.1.a

Se puede observar como una vez más la clase **Noticias** es la mejor clasificada con un 94,93%, seguido de la clase **Cuentos** (87,94%), **Monólogos** (85,83%) y por último nuevamente de la clase **Fútbol** (82,92%). No hay datos destacables con respecto a los errores, ya que estos se encuentran bastante repartidos en el resto de clases. Como curiosidad cabe comentar como un 7,12% de los trozos de la clase **Cuentos** son clasificados como **Fútbol** y viceversa, un 7,55% de los trozos de **Fútbol** son clasificados como **Cuentos**.

Comparando los resultados con los apartados anteriores, se puede apreciar como mejoran los valores de la diagonal principal: con respecto al primer apartado hay una mejora del 6,99% para la clase **Cuentos**, 3,87% para Noticias, 15,93% para **Fútbol** y un 2,76% para **Monólogos** y con respecto al segundo apartado la mejora es del 2,60% para la Clase Cuentos, 1,17% para **Fútbol** y un 3,03% para **Monólogos**. En cambio para la clase **Noticias**, el porcentaje de acierto desciende ligeramente, un 0,65% menos.

Por último, observando las desviaciones típicas halladas, se puede ver como estas son pequeñas, por lo que la concentración de datos alrededor de la media es muy alta.

KERNEL GAUSSIANO

| Trozos/Test Aleatoria/ Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 78,54 ± 2,15 | 1,24 ± 0,54 | 10,52 ± 1,25 | 9,7 ± 0,97 |
| Noticias | 1,44 ± 0,47 | 90,34 ± 1,40 | 0,80 ± 0,37 | 7,42 ± 1,33 |
| Fútbol | 10,23 ± 1,37 | 1,33 ± 0,50 | 63,16 ± 1,29 | 25,28 ± 0,89 |
| Monólogos | 5,30 ± 0,93 | 2,78 ± 0,67 | 2,52 ± 0,83 | 89,40 ± 0,81 |

Matriz de confusión 3.3.2.a

En la matriz de confusión del Kernel Gaussiano de este apartado se ve como tanto la clase **Noticias** como la clase **Monólogos** son las mejores clasificadas (porcentaje de aciertos de 90,34% y 89,4% respectivamente). Les sigue la clase **Cuentos** (78,54%), y por último, como en los apartados anteriores, la clase **Fútbol**, con solo un 63,16% de aciertos.

Comparando los aciertos con los obtenidos en los dos experimentos anteriores, se puede comprobar como mejoran los resultados con respecto al primer apartado: la clase **Cuentos** obtiene un 11,59% más de aciertos, **Noticias** un 2,99% más, **Fútbol** un 16,42% y **Monólogos** un 1,8%. En cambio, realizando la comparativa con el segundo apartado apenas mejoran los resultados: las clases **Cuentos** y **Monólogos** aumentan ligeramente, un 0,66% y un 0,34% respectivamente. Por el contrario **Noticias** y **Fútbol** consiguen un porcentaje menor de aciertos: un 1,56% y un 1,57% menos.

Al igual que sucedió utilizando este Kernel en el Experimento 2, existe un porcentaje elevado de trozos de la clase **Fútbol** que se clasifica como **Monólogos**, y sin embargo, al contrario no se confunden las clases, puesto que apenas un 2,52% de los trozos de **Monólogos** son clasificados como **Fútbol**. También vemos como nuevamente 10,52% de los trozos de la clase **Cuentos** son clasificados como **Fútbol** y viceversa en un 10,23%.

Observando ahora las desviaciones típicas se puede ver como son valores pequeños lo que significa que todos los datos se concentran alrededor de las medias mostradas en la matriz anterior. Comparando las desviaciones con las obtenidas en el Experimento 1, vemos como en este apartado son menores, lo que confirma que los resultados conseguidos en este experimento son mejores.

Conclusiones experimento 3:

Con ambos Kernels la clase mejor clasificada es **Noticias** y la peor **Fútbol**, como ya pasó en los apartados anteriores. En el Kernel Gaussiano se obtienen mejores resultados con respecto a la clase **Monólogos**, pero teniendo en cuenta las cuatro clases, una vez más el Kernel Lineal obtiene mejores resultados, sobretodo en la clase **Fútbol**, con la que se obtiene un 31,2% de aciertos más que con el Kernel Gaussiano.

4.4.3.5 Experimento 4

4.3.3.5.1 Sin clase Cuentos

KERNEL LINEAL

| Trozos/Test | Noticias | Fútbol | Monólogos |
|-------------|--------------|--------------|--------------|
| Cuentos | 8,15 ± 2,15 | 41,07 ± 9,35 | 50,78 ± 9,72 |
| Noticias | 92,58 ± 4,48 | 1,75 ± 0,73 | 5,67 ± 4,48 |
| Fútbol | 2,27 ± 0,85 | 80,68 ± 7,00 | 17,05 ± 7,00 |
| Monólogos | 3,46 ± 1,83 | 8,18 ± 2,36 | 88,36 ± 2,74 |

Matriz de confusión 3.4.1.a

Interesa ver como se reparten los trozos de la clase **Cuentos** en las otras tres clases sin meter sus datos de entrenamiento. Se observa que los trozos recaen principalmente en la clase **Monólogos** (50,78%), aunque también existe un importante porcentaje (41,07%) que se clasifica como **Fútbol**. Viendo las desviaciones típicas se puede afirmar que la clase **Monólogos** continúa siendo sobre la que más recaen los trozos de la clase **Cuentos**.

Con respecto al resto de clases se ve como se están clasificando de manera correcta un porcentaje elevado de trozos, incluso la clase **Fútbol** que es la que más problemas ha tenido en apartados anteriores obtiene un porcentaje significativo.

| Audios/Test/ | Noticias | Fútbol | Monólogos |
|--------------|----------|---------|-----------|
| Cuentos | 10 ± 32 | 41 ± 26 | 49 ± 27 |
| Noticias | 89 ± 31 | 10 ± 32 | 1 ± 2 |
| Fútbol | 2 ± 5 | 83 ± 30 | 15 ± 26 |
| Monólogos | 4 ± 13 | 2 ± 6 | 94 ± 19 |

Matriz de confusión 3.4.2.a

Los porcentajes hallados por audios son semejantes a los obtenidos por trozos. Casi la mitad de los audios son clasificados como **Monólogos** (49%), la otra mitad se reparte entre la clase **Fútbol** (41%) y **Noticias** (10%). Llamen la atención los valores elevados de las desviaciones típicas. Se confirma que la clase **Cuentos** es clasificada principalmente como **Monólogos** y **Fútbol**.

KERNEL GAUSSIANO

| Trozos/Test | Noticias | Fútbol | Monólogos |
|-------------|------------------|-------------------|-------------------|
| Cuentos | $3,78 \pm 1,48$ | $32,87 \pm 14,82$ | $63,35 \pm 15,19$ |
| Noticias | $88,27 \pm 6,38$ | $0,86 \pm 0,42$ | $10,87 \pm 6,17$ |
| Fútbol | $2,23 \pm 1,86$ | $60,93 \pm 9,84$ | $36,84 \pm 10,38$ |
| Monólogos | $3,53 \pm 1,55$ | $4,63 \pm 3,06$ | $91,84 \pm 2,64$ |

Matriz de confusión 3.4.3.a

Utilizando el Kernel Gaussiano se puede observar como los trozos de la clase **Cuentos** se clasifican principalmente como **Monólogos** en un 63,35%. El resto de los trozos recaen sobre la clase **Fútbol** en un 32,87% y en la clase **Monólogos** en un 3,78%. Con las desviaciones típicas se ve como los trozos siguen clasificándose mayoritariamente como **Monólogos**.

Observando el resto de clases se contempla como un porcentaje elevado de trozos se clasifican de forma correcta, la que más problemas sigue teniendo es la clase **Fútbol**, donde un 36,84% de los trozos son clasificados como **Monólogos**.

| Audios/Test | Noticias | Fútbol | Monólogos |
|-------------|-------------|-------------|-------------|
| Cuentos | 10 ± 32 | 37 ± 26 | 53 ± 29 |
| Noticias | 88 ± 32 | 10 ± 32 | 2 ± 6 |
| Fútbol | 2 ± 6 | 68 ± 28 | 30 ± 23 |
| Monólogos | 2 ± 6 | 0 | 98 ± 6 |

Matriz de confusión 3.4.4.a

Se puede ver como el 53% de los audios han sido clasificados como **Monólogos** y el 37% como **Fútbol**, por lo que aunque los porcentajes son más pequeños con respecto a la clasificación por trozos, coincide el hecho de que los **Cuentos** son clasificados como **Monólogos** principalmente.

Conclusiones sin incluir clase Cuentos:

Teniendo en cuenta los datos obtenidos en los dos kernels podemos afirmar que cuando no se incluye en los datos de entrenamiento la clase **Cuentos**, los trozos y los audios de dicha clase son clasificados principalmente como **Monólogos**, obteniendo la clase **Fútbol** también porcentajes a tener en cuenta.

4.3.3.5.2 Sin clase Noticias

KERNEL LINEAL

| Trozos/Test | Cuentos | Fútbol | Monólogos |
|-------------|---------------------|----------------------|----------------------|
| Cuentos | 82,58 ± 5,69 | 10,62 ± 6,94 | 6,80 ± 2,73 |
| Noticias | 13,70 ± 4,10 | 15,32 ± 11,01 | 70,98 ± 10,02 |
| Fútbol | 19,34 ± 23,25 | 66,98 ± 21,05 | 13,68 ± 5,93 |
| Monólogos | 6,54 ± 2,48 | 7,16 ± 1,72 | 86,3 ± 3,86 |

Matriz de confusión 3.4.5.a

No incluyendo los datos de entrenamiento de las **Noticias** hace que la mayoría de los trozos de esta clase caigan sobre la clase **Monólogos** (70,98%), y el resto de los trozos se repartan entre **Cuentos** y **Fútbol**. Las desviaciones típicas no aportan cambios.

| Audios/Test | Cuentos | Fútbol | Monólogos |
|-------------|----------------|----------------|----------------|
| Cuentos | 90 ± 14 | 10 ± 14 | 0 |
| Noticias | 2 ± 6 | 5 ± 11 | 93 ± 12 |
| Fútbol | 8 ± 7 | 84 ± 12 | 8 ± 10 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.6.a

Los audios de la clase **Noticias** se clasifican también mayoritariamente como **Monólogos** (93% de los audios). El porcentaje es mayor al que obteníamos en la clasificación por trozos, dejando un porcentaje muy pequeño para las otras dos clases. Teniendo en cuenta las desviaciones típicas, la clase **Noticias** sigue clasificándose como **Monólogos** en la mayor parte de los casos.

KERNEL GAUSSIANO

| Trozos/Test | Cuentos | Fútbol | Monólogos |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 71,00 ± 4,84 | 12,26 ± 10,33 | 16,74 ± 10,13 |
| Noticias | 6,35 ± 3,94 | 7,58 ± 6,67 | 86,07 ± 5,48 |
| Fútbol | 11,85 ± 4,68 | 54,67 ± 8,75 | 33,48 ± 8,18 |
| Monólogos | 5,24 ± 3,24 | 3,60 ± 2,43 | 91,16 ± 5,14 |

Matriz de confusión 3.4.7.a

Con el Kernel Gaussiano se puede ver como la mayoría de los trozos (86,07%) de la clase **Noticias** se clasifican como **Monólogos**, y el resto de trozos se reparten entre las otras dos clases, **Cuentos** y **Fútbol**. Teniendo en cuenta, a parte de las medias, las desviaciones típicas, los datos no varían.

| Audios/Test | Cuentos | Fútbol | Monólogos |
|-------------|----------------|----------------|----------------|
| Cuentos | 77 ± 21 | 20 ± 23 | 3 ± 7 |
| Noticias | 0 | 11 ± 31 | 89 ± 31 |
| Fútbol | 6 ± 5 | 60 ± 25 | 34 ± 27 |
| Monólogos | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 3.4.8.a

En la clasificación por audios se ve como las **Noticias** también son clasificadas como **Monólogos** en la mayor parte de los audios (89% de los audios). En este caso ningún audio se clasifica como **Cuentos**, y si existe una minoría (11% de los audios) que es clasificada como **Fútbol**. Al igual que pasaba en la clasificación por audios, teniendo en cuenta las desviaciones los resultados comentados no cambian.

Conclusiones sin incluir clase Noticias:

Teniendo en cuenta los datos obtenidos en los dos kernels se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Noticias**, los trozos y los audios de dicha clase son clasificados como **Monólogos** en un porcentaje muy elevado de los casos.

4.3.3.5.3 Sin clase Fútbol

KERNEL LINEAL

| Trozos/Test | Cuentos | Noticias | Monólogos |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 87,97 ± 4,58 | 1,18 ± 0,64 | 10,85 ± 4,82 |
| Noticias | 2,44 ± 1,20 | 91,74 ± 4,58 | 5,82 ± 4,80 |
| Fútbol | 32,45 ± 5,42 | 7,93 ± 2,55 | 59,62 ± 7,10 |
| Monólogos | 7,51 ± 2,86 | 3,07 ± 1,70 | 89,42 ± 2,61 |

Matriz de confusión 3.4.9.a

Utilizando el kernel lineal, los trozos de la clase **Fútbol** se clasifican principalmente como **Monólogos** (59,62%), seguido de **Cuentos** (32,45%) y **Noticias** (7,93%). Las desviaciones típicas no varían las conclusiones sacadas anteriormente.

| Audios/Test | Cuentos | Noticias | Monólogos |
|-------------|---------------|---------------|----------------|
| Cuentos | 98 ± 6 | 0 | 2 ± 6 |
| Noticias | 0 | 99 ± 2 | 1 ± 2 |
| Fútbol | 19 ± 13 | 0 | 81 ± 13 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.10.a

Realizando la clasificación por audios se puede ver como se clasifican mayoritariamente como **Monólogos** en un porcentaje superior al obtenido por trozos (81% de los audios). Tan solo un 19% son clasificados como **Cuentos**, y en este caso ninguno es clasificado como **Noticias**. Tanto en **Cuentos** como **Noticias** aparece una dispersión del 13% que no hace que varíen los datos obtenidos ni las conclusiones sacadas. Utilizando solo los datos de entrenamiento, el porcentaje de audios de **Fútbol** clasificados como **Monólogos** es superior, disminuyendo el porcentaje de la clase **Cuentos** y apareciendo un ligero porcentaje en la clase **Noticias**.

KERNEL GAUSSIANO

| Trozos/Test | Cuentos | Noticias | Monólogos |
|-------------|----------------------|----------------------|----------------------|
| Cuentos | 64,68 ± 32,90 | 6,35 ± 9,77 | 28,97 ± 23,36 |
| Noticias | 17,86 ± 34,32 | 71,51 ± 37,33 | 10,63 ± 6,40 |
| Fútbol | 19,15 ± 9,68 | 14,43 ± 22,13 | 66,42 ± 13,89 |
| Monólogos | 5,93 ± 4,06 | 3,28 ± 1,62 | 90,79 ± 4,13 |

Matriz de confusión 3.4.11.a

Con el Kernel Gaussiano se ve como la mayoría de los trozos de la clase **Fútbol** se clasifican también como **Monólogos** con un porcentaje del 66,42 %. El resto de trozos se reparten entre la clase **Cuentos** (19,15%) y la clase **Noticias** (14,43%). Observando las desviaciones típicas no hay cambios con respecto a las conclusiones sacadas, destaca sobretudo la dispersión de la clase **Noticias** por ser la más elevada.

| Audios/Test | Cuentos | Noticias | Monólogos |
|-------------|---------------|---------------|---------------|
| Cuentos | 95 ± 7 | 0 | 5 ± 7 |
| Noticias | 0 | 98 ± 6 | 2 ± 6 |
| Fútbol | 8 ± 6 | 0 | 92 ± 6 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.12.a

Realizando la clasificación por audios de la clase **Fútbol**, el mayor porcentaje de los mismos lo recibe la clase **Monólogos**, al igual que sucedía en la clasificación por trozos, solo que por audios el porcentaje es superior, ya que asciende a un 92%, siendo el resto clasificado como **Cuentos**. Las desviaciones típicas no varían las deducciones obtenidas.

Conclusiones sin incluir clase Fútbol:

Teniendo en cuenta los datos obtenidos en los dos kernels se puede afirmar que cuando no se incluye en los datos de entrenamiento la clase **Fútbol**, los trozos y los audios de dicha clase son clasificados como **Monólogos** en un porcentaje muy elevado de los casos. En segundo lugar, los datos de la clase **Fútbol** se clasifican como **Cuentos**, a excepción de la clasificación por trozos del Kernel Gaussiano, donde el segundo porcentaje más elevado se lo lleva la clase **Noticias**. El Kernel Gaussiano es el que obtiene mayor porcentaje tanto clasificando por trozos (66,42%) como por audios (92%).

4.3.3.5.4 Sin clase Monólogos

KERNEL LINEAL

| Trozos/Test | Cuentos | Noticias | Fútbol |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 86,02 ± 6,08 | 1,47 ± 0,77 | 12,51 ± 6,60 |
| Noticias | 2,63 ± 1,07 | 94,44 ± 1,20 | 2,93 ± 0,62 |
| Fútbol | 16,14 ± 7,88 | 3,12 ± 1,29 | 80,74 ± 8,45 |
| Monólogos | 30,81 ± 6,16 | 19,72 ± 1,77 | 49,47 ± 6,14 |

Matriz de confusión 3.4.13.a

Para el Kernel Lineal los trozos de la clase **Monólogos** son clasificados como **Fútbol** principalmente (un 49,47%), seguidos de las clases **Cuentos** (30,81%) y **Noticias** (19,72%). Se observa como las desviaciones típicas de las clases **Cuentos** y **Noticias** son bastante similares, teniéndolas en cuenta se ve como la clase **Monólogos** sigue clasificándose mayoritariamente como locuciones de **Fútbol**.

| Audios/Test | Cuentos | Noticias | Fútbol |
|-------------|----------------|------------|----------------|
| Cuentos | 89 ± 14 | 0 | 11 ± 14 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 11 ± 9 | 0 | 89 ± 9 |
| Monólogos | 22 ± 12 | 9 ± 7 | 69 ± 12 |

Matriz de confusión 3.4.14.a

En la clasificación por audios se ve como los **Monólogos** son clasificados principalmente como **Fútbol**, al igual que ocurría por trozos, aunque con un porcentaje más elevado (69% de los audios). Las clases **Cuentos** y **Noticias** obtienen un porcentaje del 22% y del 9% respectivamente. Las desviaciones típicas de las clases **Cuentos** y **Fútbol** coinciden, y se puede ver que no cambia la conclusión extraída indicando que los audios de la clase **Monólogos** son clasificados como **Fútbol** en la mayoría de los casos.

KERNEL GAUSSIANO

| Trozos/Test | Cuentos | Noticias | Fútbol |
|-------------|---------------------|---------------------|---------------------|
| Cuentos | 80,68 ± 8,04 | 2,54 ± 1,67 | 16,78 ± 9,16 |
| Noticias | 2,31 ± 0,98 | 94,51 ± 2,09 | 3,18 ± 1,91 |
| Fútbol | 16,44 ± 7,75 | 3,98 ± 1,79 | 79,58 ± 7,96 |
| Monólogos | 21,31 ± 5,86 | 19,90 ± 3,14 | 58,79 ± 5,36 |

Matriz de confusión 3.4.15.a

Con el Kernel Gaussiano se puede ver como al igual que ocurría en el lineal, los trozos de la clase **Monólogos** son clasificados como **Fútbol** de forma mayoritaria, y el resto de trozos están repartidos entre las clases **Cuentos** y **Noticias** (21,31% y 19,90% respectivamente). Las desviaciones no modifican la clasificación comentada anteriormente.

| Audios/Test/ | Cuentos | Noticias | Fútbol |
|--------------|----------------|------------|---------------|
| Cuentos | 79 ± 24 | 0 | 21 ± 24 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 11 ± 8 | 0 | 89 ± 8 |
| Monólogos | 9 ± 6 | 6 ± 5 | 85 ± 5 |

Matriz de confusión 3.4.16.a

Por audios la clasificación coincide con las anteriores, los audios de la clase **Monólogos** son clasificados principalmente como **Fútbol** (85% de los audios), y los otros audios se reparten entre **Cuentos** y **Noticias** (9% y 6% respectivamente). Las desviaciones pequeñas se encuentran bastante próximas a la media y no modifican las conclusiones sacadas anteriormente.

Conclusiones sin incluir clase Monólogos:

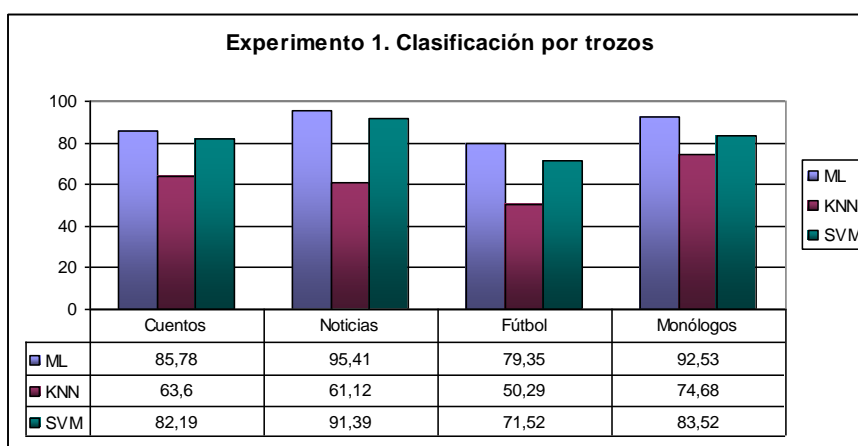
Teniendo en cuenta los datos obtenidos en los dos kernels se puede confirmar que cuando no se incluye en los datos de entrenamiento la clase **Monólogos**, los trozos y los audios de dicha clase son clasificados como si fueran de la clase **Fútbol**. El Kernel Gaussiano es el que obtiene mayor porcentaje tanto en la clasificación por trozos como por audios (58,79% y 85% respectivamente). Podemos ver como los dos kernels coinciden también en que los datos que no se clasifican como locuciones de **Fútbol** se clasifican como **Cuentos** en un porcentaje elevado de casos. El porcentaje más pequeño se lo lleva la clase **Noticias**.

4.5 Comparativa de los tres clasificadores

Por último se realizará una comparativa de los resultados conseguidos por los tres tipos de clasificadores en cada experimento realizado. Dentro de los clasificadores ML y SVM se buscaron distintas alternativas para realizar la clasificación, por este motivo se han elegido las opciones en las que se obtuvieron mejores resultados para llevar a cabo esta comparativa: en el ML el detector Multidimensional y en el SVM el Kernel Lineal.

Se representarán los datos mediante gráficos de barras en los que únicamente serán incluidos los porcentajes de acierto obtenidos.

4.5.1 Experimento 1



| Desviación típica | Cuentos | Noticias | Fútbol | Monólogos |
|-------------------|---------|----------|--------|-----------|
| ML | 6,52 | 2,33 | 7,51 | 3,09 |
| KNN | 4,14 | 7,40 | 5,95 | 4,62 |
| SVM | 5,68 | 4,16 | 8,56 | 3,80 |

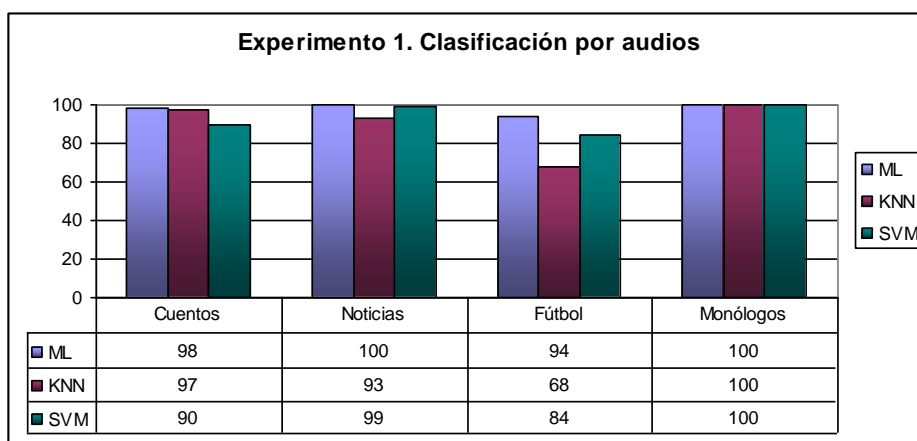
Gráfico 1

Recordemos que en este experimento se utilizaban 10 audios de cada clase para entrenar y otros 10 audios para test. En el Gráfico 1 se puede observar como al realizar la clasificación por trozos, los tres clasificadores coinciden al haber obtenido los peores resultados con la clase **Fútbol**. En cambio, con respecto a la clase mejor clasificada hay discrepancias entre los tres: la que ha conseguido mayor porcentaje de aciertos tanto en el ML como en el SVM ha sido la clase **Noticias**, y en el KNN la clase **Monólogos**.

El decisor que consigue clasificar más trozos correctamente es el ML y el que menos el KNN. Teniendo en cuenta las medias obtenidas, por trozos la clase **Noticias** es la mejor clasificada con un 95,41%, seguida por la clase **Monólogos** con un 92,53%, **Cuentos** con un 85,78% y por último, **Fútbol** con un 79,35%. Observando las desviaciones típicas de los tres clasificadores podemos ver como algunos de los resultados obtenidos para las clases **Cuentos**, **Noticias** y **Fútbol** en el SVM se solapan con los obtenidos en el ML.

Aunque teniendo en cuenta las desviaciones típicas el Kernel obtenga mejores resultados en algunas de las repeticiones realizadas, afirmaremos que el clasificador que logra porcentajes más altos continúa siendo el ML, por dos motivos:

1. Las medias son mejores (lo que se traduce en más aciertos).
2. Las desviaciones típicas son menores para todas las clases excepto para **Cuentos**, por lo que los resultados se encuentran más cerca de los obtenidos en media.



| Desviación típica | Cuentos | Noticias | Fútbol | Monólogos |
|-------------------|---------|----------|--------|-----------|
| ML | 4 | 0 | 8 | 0 |
| KNN | 5 | 10 | 68 | 100 |
| SVM | 14 | 2 | 12 | 0 |

Gráfico 2

Repitiendo el Experimento 1 pero realizando la clasificación por audios, se puede observar como siguen coincidiendo los tres clasificadores al tener los peores resultados en la clase **Fútbol**. También coinciden en que los mejores resultados los consigue la clase **Monólogos**, ya que los tres tipos de clasificadores logran acertar de qué clase es cada uno de los 10 monólogos clasificados. Las clases **Cuentos** y **Noticias** logran unos porcentajes de acierto bastante notables. En términos generales, el clasificador que consigue los mejores resultados para las cuatro clases continúa siendo el clasificador ML.

Centrándonos ahora en el clasificador ML se puede ver como todos los resultados conseguidos mejoran con respecto a la clasificación por trozos. El 100% de los audios de **Noticias** y **Monólogos** logran clasificarse de forma correcta. El porcentaje de error a la hora de clasificar la clase **Cuentos** es muy pequeño, logrando clasificar un 98% de los audios, y consigue que un 94% de los audios de **Fútbol** se clasifiquen también adecuadamente.

Teniendo en cuenta también las desviaciones típicas se puede ver como, al igual que ocurría en la clasificación por trozos, algunos resultados obtenidos con el Kernel se solapan con los conseguidos con el ML. No obstante, el clasificador ML sigue siendo el que mejores resultados obtiene, por los dos motivos que ya señalamos en la clasificación por trozos: las medias obtenidas son mejores, y las desviaciones típicas son más pequeñas, por lo que todos los resultados alcanzados en las 10 repeticiones realizadas se hallan próximas a la media.

4.5.2 Experimento 2

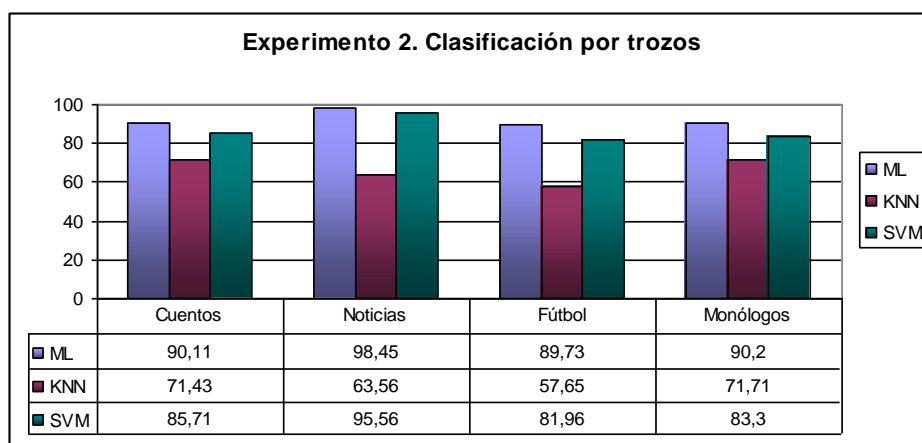


Gráfico 3

Usando la mitad de cada audio para entrenar y la otra mitad para test, observamos como el clasificador que obtiene mejores resultados continúa siendo el ML. La clase Noticias vuelve a ser la clase mejor clasificada con un 98,45%. La clase **Fútbol** en cambio, aunque es la clase con el porcentaje más bajo de aciertos (89,73%), está muy cerca de igualar el porcentaje conseguido en las clases **Cuentos** (90,11%) y **Monólogos** (90,2%).

Comparando los resultados con respecto a los obtenidos en el Experimento 1, podemos ver como mejoran los porcentajes de los tres clasificadores para las cuatro clases. Cabe destacar sobre todo el porcentaje conseguido por el clasificador ML en la clase **Fútbol**, consiguiendo un 13% más de aciertos.

4.5.3 Experimento 3

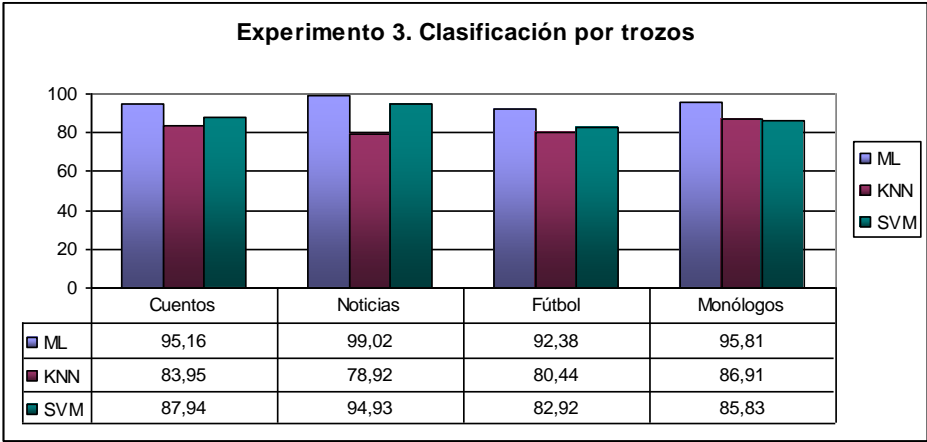


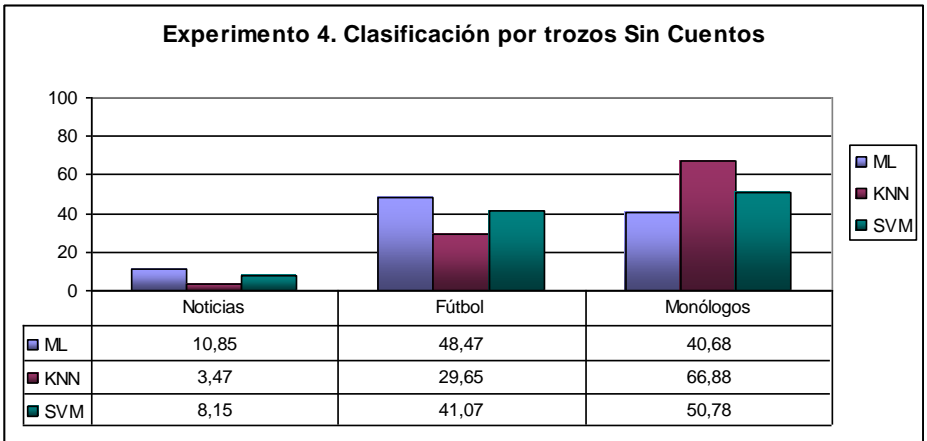
Gráfico 4

En este experimento se usaba la mitad de los audios para entrenar y la otra mitad para test al igual que en el experimento 2, pero esta vez los trozos eran cogidos de forma aleatoria. Comparando los resultados con los alcanzados en los experimentos 1 y 2 se puede observar como todas las clases consiguen mejores resultados para los tres tipos de clasificadores.

Una vez más, el clasificador ML es el que obtiene las mejores soluciones. Siendo la clase **Noticias** la mejor clasificada con un 99,02%, seguida de la clase **Cuentos** (95,16%), **Monólogos** (95,81%) y por último de la clase **Fútbol** (92,38%). Esta última logra obtener un porcentaje de acierto bastante importante, superando por primera vez el 90% de aciertos.

4.5.4 Experimento 4

Sin Cuentos

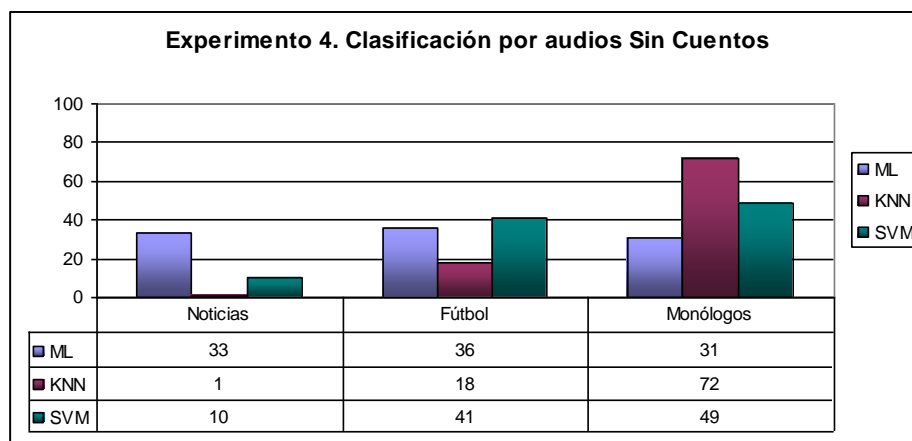


| Desviación típica | Noticias | Fútbol | Monólogos |
|-------------------|----------|--------|-----------|
| ML | 3,10 | 8,72 | 9,05 |
| KNN | 1,41 | 7,41 | 7,79 |
| SVM | 2,15 | 9,35 | 9,72 |

Gráfico 5

En el Gráfico 5 se puede ver como se reparten los trozos al no incluir los datos de la clase **Cuentos** en el entrenamiento. Utilizando el clasificador ML los trozos de la clase **Cuentos** se reparten entre la clase **Fútbol** y **Monólogos** principalmente, y haciendo uso de los clasificadores KNN y SVM se clasifican como si fueran de la clase Monólogos en la mayor parte de los casos.

Observando la distribución de los datos se podría afirmar que los trozos de la clase **Cuentos** son clasificados mayoritariamente como **Monólogos**.



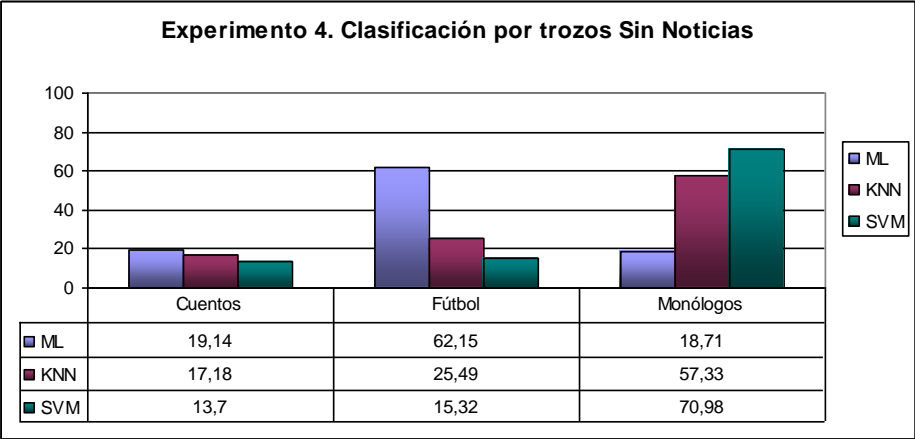
| Desviación típica | Noticias | Fútbol | Monólogos |
|-------------------|----------|--------|-----------|
| ML | 26 | 16 | 21 |
| KNN | 32 | 15 | 28 |
| SVM | 32 | 26 | 27 |

Gráfico 6

En el Gráfico 6 se observa como la clasificación por audios de la clase **Cuentos** es, a grandes rasgos, similar a la clasificación por trozos. Utilizando el clasificador ML, los audios se reparten aproximadamente a partes iguales entre las otras tres clases, siendo algo superior los clasificados como locuciones de **Fútbol**. Haciendo uso del clasificador KNN los audios son clasificados como **Monólogos**, en el caso del clasificador SVM los audios se reparten principalmente entre las clases **Fútbol** y **Monólogos**, siendo algo más elevado en esta última clase.

Observando la distribución de los datos se puede indicar que, al igual que sucedió en la clasificación por trozos, los audios de la clase **Cuentos** se clasifican principalmente como **Monólogos**.

Sin Noticias

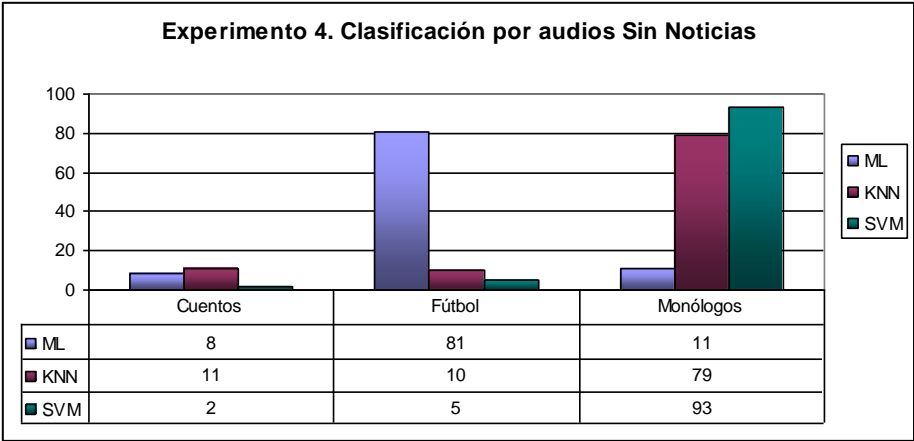


| Desviación típica | Cuentos | Fútbol | Monólogos |
|-------------------|---------|--------|-----------|
| ML | 10,92 | 19,04 | 11,65 |
| KNN | 3,78 | 5,00 | 5,09 |
| SVM | 4,10 | 11,01 | 10,02 |

Gráfico 7

En el Gráfico 7 se puede ver como se reparten los trozos al dejar fuera del entrenamiento los datos de la clase **Noticias**. Utilizando el clasificador ML los trozos de la clase **Noticias** se clasifican principalmente como si fueran de la clase **Fútbol**, y en cambio, haciendo uso de los clasificadores KNN y SVM se clasifican como si fueran de la clase **Monólogos** en un porcentaje elevado.

Observando la distribución de los datos se puede ver como en este caso hay más discrepancias que en el caso anterior, ya que aunque se podría afirmar que los trozos de la clase **Noticias** se clasifican como **Monólogos**, al ser así como lo indican dos de los tres clasificadores utilizados, hay que tener en cuenta que el clasificador restante, el ML, es el que ha obtenido mejores resultados a lo largo de todo el trabajo, por lo que se dirá que los trozos de la clase **Noticias** son clasificados mayoritariamente como **Monólogos** y **Fútbol**.

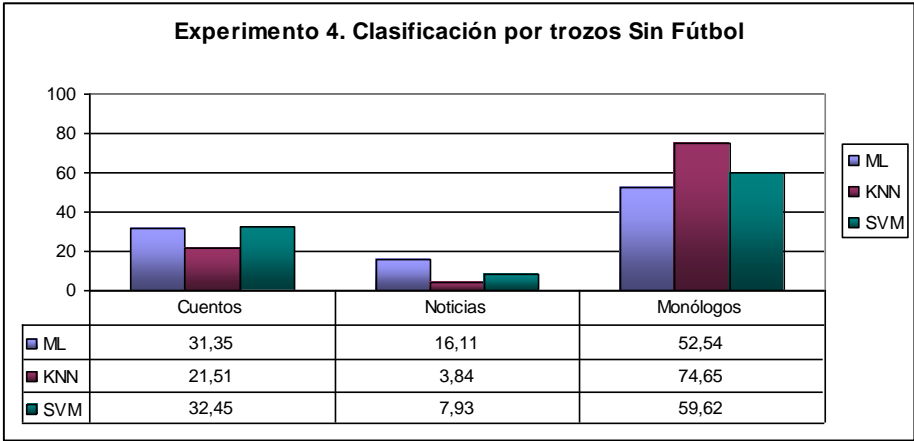


| Desviación típica | Cuentos | Fútbol | Monólogos |
|-------------------|---------|--------|-----------|
| ML | 9 | 14 | 13 |
| KNN | 16 | 15 | 8 |
| SVM | 6 | 11 | 12 |

Gráfico 8

En el Gráfico 8 se observa como la clasificación por audios de la clase **Noticias** coincide con la clasificación por trozos, ya que haciendo uso del clasificador ML los audios son clasificados como locuciones de Fútbol y en cambio utilizando los clasificadores KNN y SVM son clasificados como **Monólogos**.

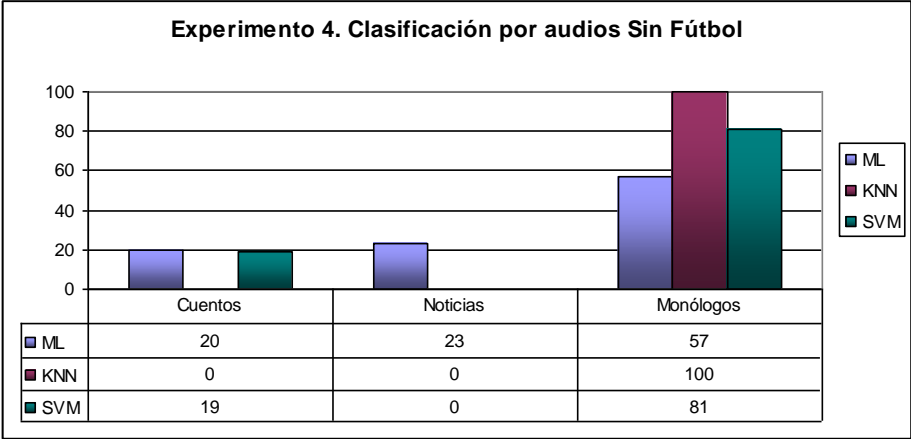
Sin Fútbol



| Desviación típica | Cuentos | Noticias | Monólogos |
|-------------------|---------|----------|-----------|
| ML | 9,41 | 3,66 | 10,59 |
| KNN | 2,50 | 0,93 | 2,64 |
| SVM | 5,42 | 2,55 | 7,10 |

Gráfico 9

En el Gráfico 9 se ve como se clasifican los trozos al no incluir los datos de la clase **Fútbol** dentro del entrenamiento. En este caso hay unanimidad por parte de los tres clasificadores, ya que coinciden al clasificar estos trozos como si fueran de la clase **Monólogos** principalmente. También coinciden en que en segundo lugar son clasificados como **Cuentos**.

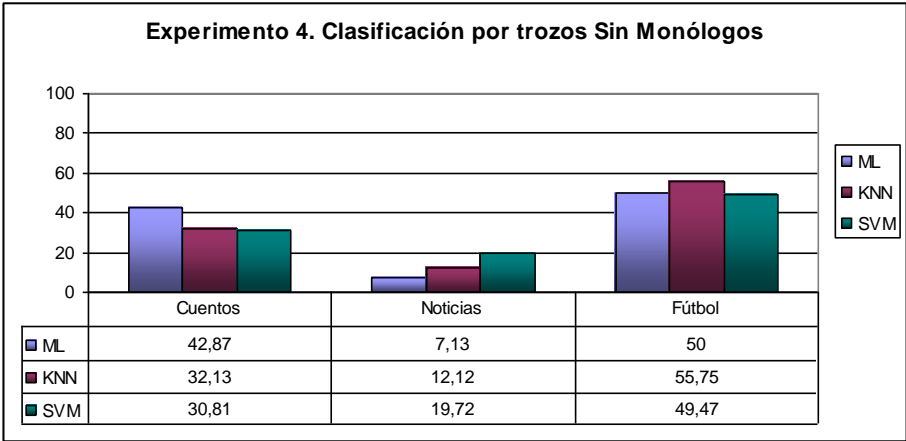


| Desviación típica | Cuentos | Noticias | Monólogos |
|-------------------|---------|----------|-----------|
| ML | 21 | 13 | 20 |
| KNN | 0 | 0 | 100 |
| SVM | 13 | 0 | 13 |

Gráfico 10

El Gráfico 10 no hace sino confirmar los resultados obtenidos en la clasificación por trozos, ya que la mayoría de los audios de la clase **Fútbol** son clasificados como **Monólogos** al no incluir información de estos dentro de los datos de entrenamiento.

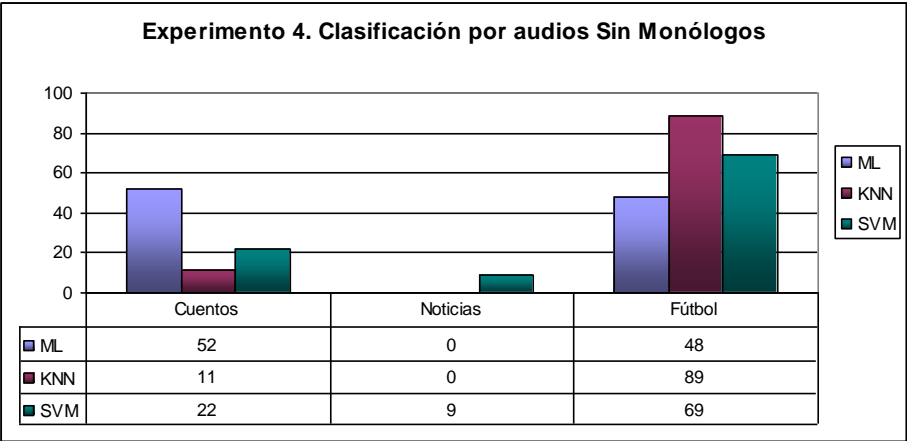
Sin Monólogos



| Desviación típica | Cuentos | Noticias | Fútbol |
|-------------------|---------|----------|--------|
| ML | 13,34 | 1,20 | 12,66 |
| KNN | 6,73 | 1,91 | 5,75 |
| SVM | 6,16 | 1,77 | 6,14 |

Gráfico 11

Observando la distribución de los trozos de la clase **Monólogos** en el Gráfico 11, se puede ver como estos se reparten principalmente entre las clases **Cuentos** y **Fútbol** para los tres clasificadores, siendo los porcentajes mayores con respecto a la clase **Fútbol**.



| Desviación típica | Cuentos | Noticias | Fútbol |
|-------------------|---------|----------|--------|
| ML | 15 | 0 | 15 |
| KNN | 15 | 0 | 15 |
| SVM | 12 | 7 | 12 |

Gráfico 12

En el Gráfico 12 se puede ver como los audios de la clase **Monólogos** se clasifican como locuciones de **Fútbol** utilizando los clasificadores KNN y SVM. En el caso del clasificador ML, los audios se reparten entre las clases **Cuentos** y **Fútbol**. No obstante, observando la distribución de los audios se dirá que estos son clasificados principalmente como si fueran audios de **Fútbol**, al igual que sucedía en la clasificación por trozos.

CAPÍTULO 5 CONCLUSIONES Y LÍNEAS FUTURAS

En este último capítulo se hará un breve resumen del trabajo realizado para centrarnos posteriormente en las conclusiones extraídas a partir del análisis de los resultados. Finalmente se expondrán posibles líneas futuras relacionadas con el estudio desarrollado en este proyecto.

5.1 Conclusiones

Lo primero que se hizo fue crear una **base de datos** de audio de cuatro géneros de narración diferentes. Se eligieron cuatro estilos distintos entre sí pero que a la vez tuvieran similitudes para poner a prueba el clasificador diseñado: Cuentos, Noticias, Fútbol y Monólogos.

Una vez creada la base de datos, se programaron diferentes clasificadores para separar los géneros. Se seleccionaron tres tecnologías para llevar a cabo la clasificación: de Máxima Verosimilitud (ML), Vecino más cercano (KNN) y Máquina de Vectores de Soporte (SVM), dentro de las cuales, a su vez, se barajaron distintas opciones para poder decidir cuál era la más apta al conseguir los mejores resultados. La implementación de estos clasificadores fue realizada con el programa matemático **Matlab**. Por último se realizaron diferentes experimentos para poder analizar en detalle las prestaciones de cada uno de los clasificadores.

Como se ha indicado previamente, dentro de cada tecnología se barajaron distintas alternativas. En el caso del ML las opciones fueron: detector **Identidad**, detector **Diagonal** y detector **Multidimensional**. Se pudo comprobar que el detector Diagonal mejoraba los datos conseguidos con el detector Identidad en un 14,57% y que a su vez el detector Multidimensional lograba aumentar los porcentajes de acierto de los dos anteriores. La mejora con respecto al detector Diagonal fue de un 5,21% y en relación al detector Identidad de un 20,54%. Se recordará que el detector Multidimensional era en el que se pensaba que el conjunto de datos de entrenamiento provenía de una gaussiana multidimensional, es decir aquella que usaba covarianzas completas. Que los resultados sean superiores con el detector Multidimensional puede deberse a que al ser una matriz completa, con datos en todas sus filas y columnas, hace que sea capaz de capturar todas las relaciones y las explota para mejorar los resultados.

En la tecnología SVM los mejores resultados fueron obtenidos con el Kernel Lineal frente al Kernel Gaussiano. El porcentaje de mejora del Kernel Lineal con respecto al Kernel Gaussiano fue de un 9,47%.

Teniendo en cuenta las consideraciones anteriores, para realizar la comparación entre los tres tipos de clasificadores se utilizaron los resultados conseguidos en el ML con el detector Multidimensional, SVM con el Kernel Lineal y el KNN.

El clasificador ML obtuvo los mejores porcentajes de acierto para las cuatro clases. El acierto total utilizando este método fue de un 88,27%, el cuál fue calculado hallando la media de los porcentajes de acierto conseguidos por las cuatro clases. Con respecto a la siguiente tecnología utilizada, SVM, obtuvo un porcentaje de mejora del 7,45%. La tecnología que consiguió los peores resultados para cada una de las cuatro clases es KNN. El porcentaje de mejora del ML con respecto al KNN fue bastante apreciable: un 41,41%.

Tal y como se ha comentado en el párrafo anterior, el clasificador que obtuvo los peores resultados y por lo tanto cometió más errores a la hora de clasificar las cuatro clases es el KNN. El porcentaje de acierto total del KNN fue de un 62,42%. Por audios completos en cambio el porcentaje mejoró, ya que consiguió clasificar correctamente un 89,5% de los audios, es decir, un 43,38% más que por trozos, un porcentaje de mejora notable.

En general se pudo comprobar que realizando la clasificación por audios completos en vez de por trozos individuales los resultados mejoraban considerablemente. Haciendo uso del clasificador ML por audios se consiguieron clasificar correctamente un 98%, es decir, un 11,02% más que por trozos.

En los experimentos 2 y 3 se pudo constatar como los porcentajes de acierto iban en aumento al utilizar trozos del audio a clasificar dentro del entrenamiento. Utilizando la primera mitad del audio para entrenar y la segunda para test, los resultados ya mejoraban con respecto a utilizar audios completos, pero utilizando partes aleatorias el porcentaje de mejora aumenta aún más. Esto era algo previsible ya que si se cogen trozos de distintas partes del audio, resulta más fácil que este pueda clasificar los datos de test ya que habrá partes muy similares. En cambio de la primera mitad a la segunda el audio varía y el sistema puede cometer más errores. No obstante, aunque el sistema progresa, no existe una gran diferencia entre los porcentajes de acierto, por lo que no sería necesario utilizar una base de datos más amplia, aunque sería interesante comprobarlo para ver como varían los resultados. Tomando como ejemplo el clasificador ML, el porcentaje de mejora cogiendo la primera mitad para entrenar y la segunda para test (Experimento 2) es 4,36%, y en el caso de dividir los audios de manera aleatoria para formar los conjuntos de los datos de entrenamiento y test (Experimento 3) el porcentaje es superior, ya que alcanza el 8,29%.

Comparando los resultados de todos los clasificadores se pudo ver como todos coincidían al obtener el mejor porcentaje de acierto en la clase Noticias, a excepción del clasificador KNN y el Kernel Gaussiano, cuyo máximo porcentaje recaía sobre la clase Monólogos.

La clase Noticias consiguió un porcentaje de acierto de 95,41% en la clasificación por trozos y del 100 % en la clasificación por audios haciendo uso del ML Multidimensional. La siguiente clase que obtuvo un porcentaje destacable fue Monólogos, logrando un 92,53% de aciertos en la clasificación por trozos y un interesante 98% en la clasificación por audios.

Analizando los resultados obtenidos con el resto de clasificadores se pudo comprobar como los otros dos decisores ML, el Identidad y el Diagonal, coincidieron con el Multidimensional, así como el clasificador SVM con el Kernel Lineal, en cambio, empleando el KNN y el SVM con el Kernel Gaussiano la clase que obtuvo mayor porcentaje de aciertos fue Monólogos, aunque quedó lejos del porcentaje obtenido por Noticias, un 74,68% para el KNN y 87,81% para el SVM.

La clase Fútbol fue la clase que tuvo más problemas para ser clasificada. Todos los clasificadores coincidían en que los peores porcentajes de acierto los recibía la clase Fútbol tanto si la clasificación se realizaba por trozos o por audios. En la clasificación por trozos, todos ellos catalogaban los trozos de la clase Fútbol como si provinieran de la clase Monólogos, sobretodo en el clasificador KNN y en el SVM con el Kernel Gaussiano donde un 36,61% y un 31,86% de los trozos se clasificaban como Monólogos. En el caso del ML Multidimensional este consiguió rebajar el porcentaje a 17,29%. En el caso del Kernel Lineal los trozos de la clase Fútbol se repartieron entre Cuentos y Monólogos (12,67% y 13,80%). Teniendo en cuenta ahora la clasificación por audios, se pudo ver como todos los clasificadores continuaban teniendo problemas para clasificar esta clase, aunque los porcentajes de acierto aumentaron con respecto a la clasificación por trozos. En el caso del KNN y el SVM con Kernel Gaussiano los porcentajes de la clasificación de esta clase como Monólogos continuaron siendo elevados (25% y 24%). En el ML Multidimensional tan solo un 5% de los audios de la clase Fútbol fueron clasificados como Monólogos. La segunda clase con la que más se confundió fue la clase Cuentos, en cambio el porcentaje que recayó sobre la clase Noticias fue muy pequeño.

Algo común entre las clases Cuentos, Noticias y Fútbol fue que los porcentajes más altos de error recayeron principalmente sobre la clase Monólogos. Puede deberse a que el tono de los Monólogos es bastante variable dependiendo del tema que traten, por lo que el resto de clases pueden hallar similitudes y por lo tanto confundirse a la hora de ser clasificadas. Los errores cuando se clasificó la clase Monólogos recayeron principalmente sobre las clases Cuentos y Fútbol.

Otra forma de ver que clases se parecen más y, por lo tanto, tienen más riesgo de confundirse entre sí, es comprobando cómo se clasifican los datos de cada clase ante la ausencia de los datos de sí mismos dentro del entrenamiento. Interesa ver si los trozos de la clase ausente en el entrenamiento se reparten equitativamente entre las tres restantes o por el contrario recaen más sobre alguna de ellas. En el caso de la clase Fútbol coincidieron todos los decisores al clasificar dicha clase como Monólogos por trozos y

audios. Con el resto de clases sin embargo existieron discrepancias. Las clases Cuentos y Noticias se clasificaron como Fútbol principalmente utilizando el clasificador ML y como Monólogos con los clasificadores KNN y SVM, ambos coincidieron tanto por trozos como por audios. La clase Monólogos se clasificó también como si fuera de la clase Fútbol realizando la clasificación por trozos con los tres clasificadores, sin embargo por audios se clasificaron como Cuentos en el ML y como Fútbol en el KNN y SVM.

5.2 Líneas futuras

Se presentan numerosas opciones para continuar con el trabajo desarrollado en este proyecto. A continuación se detallan algunas de estas posibles líneas futuras de trabajo:

- Modificar la base de datos. La forma en la que se han implementado los clasificadores nos ofrece la posibilidad de repetir los experimentos utilizando otras bases de datos sin tener que modificar código, y de esa manera comprobar si efectivamente es viable crear un sistema de estas características. Las posibles modificaciones serían:
 - Ampliar la base de datos para ver si de esa manera se reduce el porcentaje de error.
 - Modificar el género del hablante, es decir, formar una base de datos en las que los locutores sean mujeres.
 - Probar con audios en otros idiomas.
 - Realizar pruebas más complejas como escoger cuatro clases muy similares, es decir, escoger un tema y que las cuatro clases estén relacionadas. Por ejemplo: deportes, y elegir distintos deportes colectivos, como podrían ser Baloncesto, Fútbol, Balonmano y Waterpolo. De esta manera se podría decidir si los locutores de los distintos deportes utilizan el mismo estilo de voz o por el contrario varían el tono.
- Emplear otro clasificador diferente a los tres desarrollados en este trabajo.
- Introducir variaciones en el diseño y en la implementación de los clasificadores para aumentar el rendimiento.
- Reprogramar el código en otro lenguaje para tratar de mejorar la velocidad de ejecución.

APÉNDICE

Los experimentos realizados con cada uno de los clasificadores llevados a cabo en este proyecto se repitieron utilizando los datos de entrenamiento tanto para entrenar como para testear la máquina para poder confirmar que el sistema no estaba sobreajustando. En este apéndice se incluyen todos los resultados en forma de matrices de confusión.

CLASIFICADOR ML

Experimento 1

DETECTOR IDENTIDAD:

| Trozos/Entrenamiento Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 86,34 ± 2,53 | 2,05 ± 0,55 | 7,15 ± 1,16 | 4,46 ± 2,47 |
| Noticias | 1,05 ± 0,42 | 91,94 ± 0,99 | 1,83 ± 0,97 | 5,18 ± 1,43 |
| Fútbol | 11,98 ± 2,89 | 4,12 ± 1,24 | 69,56 ± 4,99 | 14,34 ± 3,13 |
| Monólogos | 9,91 ± 1,49 | 5,32 ± 1,86 | 7,91 ± 0,86 | 76,86 ± 2,65 |

Matriz de confusión 1.1.1.b

Utilizando el conjunto de entrenamiento como datos de test, se observa que con respecto a la clasificación por trozos los resultados mejoran notablemente en las clases **Cuentos** y **Fútbol**, y sutilmente en la clase **Noticias**, donde el porcentaje supera el 90% de aciertos. Con respecto a la clase **Monólogos** también mejoran ligeramente los resultados aunque sigue cometiendo errores con respecto al resto de clases. Si bien los resultados de la matriz 1.1.1.b son mejores comparados con los de la matriz 1.1.1.a (como era de suponer, al utilizar los mismos datos para entrenar y testear), el porcentaje de errores no es mucho menor, por lo que podemos afirmar que el clasificador no está sobreajustando.

La desviación típica se ve suavizada en comparación con la que se veía en la clasificación por trozos, prácticamente todos los valores están próximos a la media hallada. El valor de la desviación más alto es el de la clase **Fútbol** con respecto a sí mismo.

| Audios/Entrenamiento Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------------------|---------------|------------|----------------|---------------|
| Cuentos | 94 ± 8 | 0 | 6 ± 8 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 4 ± 5 | 2 ± 4 | 84 ± 10 | 1 ± 8 |
| Monólogos | 4 ± 5 | 0 | 0 | 96 ± 5 |

Matriz de confusión 1.1.2.b

En la clasificación por audios el porcentaje de aciertos mejora en todas las clases excepto en la de **Monólogos** que desciende ligeramente. En la matriz 1.1.2.b se puede observar que la mayor dispersión es la de clase **Fútbol** consigo misma, al igual que ocurría en la clasificación por trozos. Se considera que esta mejoría está dentro de los límites de la normalidad, el sistema no sobreajusta.

DETECTOR DIAGONAL:

| Trozos/Entrenamiento Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------------------|--------------------|---------------------|---------------------|--------------------|
| Cuentos | 92,8 ± 2,56 | 1,64 ± 0,68 | 1,12 ± 1,00 | 4,44 ± 1,95 |
| Noticias | 0,82 ± 0,38 | 98,33 ± 0,30 | 0,4 ± 0,23 | 0,45 ± 0,33 |
| Fútbol | 3,43 ± 1,53 | 0,54 ± 0,52 | 88,19 ± 3,59 | 7,84 ± 2,89 |
| Monólogos | 0,94 ± 0,39 | 4,26 ± 2,01 | 8 ± 0,86 | 86,8 ± 2,31 |

Matriz de confusión 1.1.3.b

En la Matriz de confusión 1.1.3.b se puede ver como utilizando solo el conjunto de entrenamiento los resultados mejoran para las cuatro clases pero sin llegar al sobreajuste, puesto que aunque para la clase **Noticias** el porcentaje de aciertos es muy alto, para las otras tres clases aparecen porcentajes de error considerables, sobretodo en las clases **Fútbol** y **Monólogos**, las cuáles son confundidas entre sí un 7,84% y un 8% respectivamente.

Mirando las desviaciones típicas se observa como estas se ven mitigadas con respecto a las que se contemplaban en la matriz de confusión 1.1.3.a, todos los valores se encuentran próximos a la media. En la fila de la clase **Fútbol** es donde se puede ver una desviación ligeramente mayor tanto en la columna de **Fútbol** como en la de **Monólogos**.

| Audios/Entrenamiento Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------------------|------------------|------------|------------------|------------------|
| Cuentos | 99 ± 3,16 | 0 | 0 | 1 ± 3,16 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 1 ± 3,16 | 0 | 99 ± 3,16 | 0 |
| Monólogos | 0 | 0 | 2 ± 6,32 | 98 ± 6,32 |

Matriz de confusión 1.1.4.b

En la clasificación por audios de los datos de entrenamiento, mejoran todas las clases con respecto a los datos de test, disminuyendo a su vez las desviaciones típicas, con excepción de la clase **Monólogos**, (como ya pasó en el decisor Identidad) la cuál disminuye un 1,02% y la dispersión pasa de 3% a 6,32%. El resto de clases aumenta su porcentaje de aciertos de la siguiente manera: La clase **Cuentos** un 10%, la clase **Noticias** un 1,01% (clasifica todos los audios correctamente) y la clase **Fútbol** un 12,5%.

DETECTOR MULTIDIMENSIONAL:

| Trozos/Entrenamiento Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|---------------------------------------|------------|------------|------------|------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 1.1.5.b

Utilizando los datos de entrenamiento como test se clasifican correctamente todos los trozos de las cuatro clases, por lo que se podría decir que el decisor está sobreajustando, en cambio, se dirá que no es así ya que si se comparan estos porcentajes con los de la Matriz de confusión 1.1.5.a el aumento es razonable. Al clasificar todos los datos de forma correcta las dispersiones son nulas.

| Audios/Entrenamiento Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|---------------------------------------|------------|------------|------------|------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 1.1.6.b

Se advierte que, al igual que sucedió en la clasificación por trozos, la totalidad de los audios han sido correctamente clasificados utilizando únicamente los datos de entrenamiento. La desviación típica es nula.

Experimento 2

DETECTOR IDENTIDAD

| Trozos/Entrenamiento Mitad/Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|---|--------------|--------------|--------------|--------------|
| Cuentos | 83,99 | 2,53 | 9,55 | 3,93 |
| Noticias | 1,33 | 87,56 | 2,44 | 8,67 |
| Fútbol | 12,5 | 5,61 | 65,82 | 16,07 |
| Monólogos | 10,16 | 6,32 | 9,71 | 73,81 |

Matriz de confusión 1.2.1.b

Haciendo uso únicamente de los datos de entrenamiento se puede ver como los resultados no son mejores. Las clases **Cuentos** y **Fútbol** mejoran ligeramente (2,25% y 1,40%), y sin embargo las otras dos clases, **Noticias** y **Monólogos**, los porcentajes de acierto son peores (un 5,96% y un 3,37% menos). Fijándose en los valores fuera de la diagonal principal, se puede observar como crecen los errores al clasificar la clase **Noticias** como **Monólogos**. Por otra parte, en la clase **Fútbol** vuelven a disminuir los trozos clasificados como **Cuentos** y aumentan de nuevo los clasificados como **Monólogos**. Teniendo en cuenta estos datos, se afirmará que el detector no sobreajusta.

DETECTOR DIAGONAL

| Trozos/Entrenamiento Mitad/Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|--|--------------|--------------|--------------|--------------|
| Cuentos | 92,13 | 1,69 | 1,69 | 4,49 |
| Noticias | 1,11 | 97,56 | 0,44 | 0,89 |
| Fútbol | 3,06 | 1,28 | 85,71 | 9,95 |
| Monólogos | 0,45 | 6,55 | 10,38 | 82,62 |

Matriz de confusión 1.2.2.b

Del mismo modo que en el detector Identidad, aplicando únicamente los datos de entrenamiento solo mejoran las clases **Cuentos** y **Fútbol** (un 0,40% y un 4,57% respectivamente), los resultados de las clases **Noticias** y **Monólogos** son peores (un 0,89% y 3,89% menos). Observando los errores se ve como los trozos clasificados como **Cuentos** en las clases **Fútbol** y **Monólogos** se han visto reducidos. En la clase **Monólogos** aumentan los trozos clasificados como **Fútbol**.

DETECTOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Mitad/Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|--|------------|------------|------------|------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 1.2.3.b

Utilizando como datos únicamente los de entrenamiento sucede lo mismo que pasó en el experimento anterior, se clasifica correctamente la totalidad de los trozos, siendo esto comprensible ya que los porcentajes de acierto de todas las clases rondaba el 90%, e incluso la clase **Noticias** estaba en un 98,45% por lo que se puede decir que no sobreajusta.

Experimento 3

DECISOR IDENTIDAD

| Trozos/Entrenamiento Aleatorio /Identidad | Cuentos | Noticias | Fútbol | Monólogos |
|--|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 84,08 ± 1,33 | 2,13 ± 0,58 | 10,14 ± 0,82 | 3,65 ± 1,11 |
| Noticias | 1,2 ± 0,46 | 91,75 ± 0,97 | 1,42 ± 0,91 | 5,63 ± 1,21 |
| Fútbol | 13,67 ± 1,01 | 5,35 ± 0,84 | 66,62 ± 1,55 | 14,36 ± 0,85 |
| Monólogos | 10,84 ± 1,11 | 6,09 ± 0,64 | 6,73 ± 0,88 | 76,34 ± 1,48 |

Matriz de confusión 1.3.1.b

En este experimento mejoran todos los resultados con respecto a los contemplados en la matriz 1.3.1.a, aunque discretamente: la clase **Cuentos** obtiene un 2,73% más de aciertos, **Noticias** un 1,47%, **Fútbol** un 1,26% y **Monólogos** un 2,62%. Las desviaciones típicas son similares a las obtenidas utilizando los datos de test. El sistema no sobreajusta.

DECISOR DIAGONAL

| Trozos/Entrenamiento Aleatorio/Diagonal | Cuentos | Noticias | Fútbol | Monólogos |
|--|------------------|---------------------|---------------------|---------------------|
| Cuentos | 92 ± 0,71 | 1,35 ± 0,41 | 2,02 ± 0,53 | 4,63 ± 0,60 |
| Noticias | 1,02 ± 0,28 | 97,96 ± 0,42 | 0,29 ± 0,26 | 0,73 ± 0,26 |
| Fútbol | 4,31 ± 0,75 | 0,87 ± 0,37 | 86,61 ± 1,30 | 8,21 ± 0,93 |
| Monólogos | 0,95 ± 0,35 | 5,46 ± 0,64 | 8,6 ± 0,95 | 84,99 ± 1,50 |

Matriz de confusión 1.3.2.b

Haciendo uso únicamente de los datos de entrenamiento nuevamente se obtienen mejores resultados para todas las clases, aunque se puede comprobar como los porcentajes de acierto no difieren demasiado de los logrados con los datos de test. Las dispersiones halladas son ligeramente menores a las obtenidas con los datos de test. El decisor diagonal no sobreajusta.

DECISOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Aleatorio/Multidimensional | Cuentos | Noticias | Fútbol | Monólogos |
|---|------------|------------|---------------------|-------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 99,97 ± 0,08 | 0,03 ± 0,08 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 1.3.3.b

Utilizando únicamente los datos de entrenamiento se ve como a excepción de la clase **Fútbol** que tiene un pequeño porcentaje de error con respecto a la clase **Monólogos**, la totalidad de los trozos de las otras tres clases son clasificados correctamente.

Las desviaciones típicas son 0, únicamente en la clase **Fútbol** aparece una ligera dispersión. Los porcentajes utilizando los datos de test ya eran bastante elevados, por lo que aunque con los datos de entrenamiento se consigan clasificar casi toda la totalidad de los trozos de forma correcta, el sistema no sobreajusta.

Experimento 4

Sin Cuentos

DECISOR IDENTIDAD

| Trozos/Entrenamiento Identidad | Noticias | Fútbol | Monólogos |
|--------------------------------|---------------------|----------------------|--------------------|
| Cuentos | 7,85 ± 2,25 | 54,04 ± 12,23 | 38,11 ± 12,57 |
| Noticias | 92,15 ± 1,05 | 2,25 ± 0,92 | 5,60 ± 1,67 |
| Fútbol | 4,39 ± 0,88 | 79,47 ± 3,30 | 16,14 ± 3,53 |
| Monólogos | 5,38 ± 1,89 | 10,22 ± 1,06 | 84,4 ± 2,11 |

Matriz de confusión 1.4.1.b

En la clasificación utilizando solo los datos de entrenamiento, todos los porcentajes son mejores como era de esperar, y en caso de la clase **Cuentos**, no cambian los resultados obtenidos. Esto último es lógico porque de cara a la clase cuentos no han variado los datos utilizados para realizar la clasificación.

| Audios/Entrenamiento Identidad | Noticias | Fútbol | Monólogos |
|-----------------------------------|------------|-------------|--------------|
| Cuentos | 1 ± 3 | 69 ± 20 | 30 ± 19 |
| Noticias | 99 ± 3 | 1 ± 3 | 0 |
| Fútbol | 2 ± 4 | 84 ± 27 | 14 ± 28 |
| Monólogos | 1 ± 3 | 1 ± 3 | $0,98 \pm 6$ |

Matriz de confusión 1.4.2.b

En la Matriz de confusión 1.4.2b se ven incrementados los audios clasificados como **Fútbol** (69%), disminuyen los audios clasificados como **Monólogos** y aparece un pequeño porcentaje clasificado como **Noticias**.

DECISOR DIAGONAL

| Trozos/Entrenamiento Diagonal | Noticias | Fútbol | Monólogos |
|----------------------------------|------------------|------------------|------------------|
| Cuentos | $14,20 \pm 4,04$ | $42,26 \pm 8,67$ | $43,54 \pm 8,73$ |
| Noticias | $98,77 \pm 0,29$ | $0,56 \pm 0,22$ | $0,67 \pm 0,37$ |
| Fútbol | $0,59 \pm 0,59$ | $91,1 \pm 3,45$ | $8,31 \pm 3,07$ |
| Monólogos | $4,26 \pm 2,01$ | $8,38 \pm 0,90$ | $87,36 \pm 2,28$ |

Matriz de confusión 1.4.3.b

Nuevamente los resultados con respecto a la clase **Cuentos** son los mismos que los vistos en la Matriz de confusión 1.4.3.a al no haber incluido datos de dicha clase en el entrenamiento. En el resto de clases los porcentajes de acierto mejoran considerablemente.

| Audios/Entrenamiento Diagonal | Noticias | Fútbol | Monólogos |
|----------------------------------|------------|-------------|-------------|
| Cuentos | 9 ± 13 | 46 ± 21 | 45 ± 20 |
| Noticias | 100 | 0 | 0 |
| Fútbol | 0 | 99 ± 3 | 1 ± 3 |
| Monólogos | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 1.4.4.b

Comparando los resultados con los calculados utilizando los datos de test, en la clasificación por audios coincide el valor con respecto a la clase **Fútbol** (46%) pero esta vez la clase **Monólogos** se ve atenuada a un 45%, aumentando ligeramente la clase **Noticias**.

DECISOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Multidimensional | Noticias | Fútbol | Monólogos |
|---------------------------------------|--------------|--------------|--------------|
| Cuentos | 10,85 ± 3,10 | 48,47 ± 8,72 | 40,68 ± 9,05 |
| Noticias | 98,64 ± 0,31 | 0,65 ± 0,25 | 0,71 ± 0,35 |
| Fútbol | 0,54 ± 0,53 | 91,02 ± 3,72 | 8,44 ± 3,35 |
| Monólogos | 4,28 ± 1,88 | 7,87 ± 1,01 | 87,85 ± 2,11 |

Matriz de confusión 1.4.5.b

Haciendo uso solo datos de entrenamiento se obtienen los mismos resultados que en la clasificación usando también los datos de test para la clase Cuentos y en el resto de clases los porcentajes se ven aumentados.

| Audios/Entrenamiento Multidimensional | Noticias | Fútbol | Monólogos |
|---------------------------------------|----------|---------|-----------|
| Cuentos | 33 ± 31 | 29 ± 14 | 38 ± 20 |
| Noticias | 100 | 0 | 0 |
| Fútbol | 0 | 99 ± 3 | 1 ± 3 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.6.b

Utilizando únicamente los datos de entrenamiento los audios se reparten de distinta manera: la clase con mayor porcentaje de acierto es **Monólogos** (38% de audios), seguida de **Noticias** (33%) y **Fútbol** (29%).

Sin Noticias

DECISOR IDENTIDAD

| Trozos/Entrenamiento Identidad | Cuentos | Fútbol | Monólogos |
|--------------------------------|--------------|---------------|---------------|
| Cuentos | 87,27 ± 2,17 | 7,91 ± 1,58 | 4,82 ± 2,35 |
| Noticias | 6,91 ± 5,14 | 39,77 ± 22,16 | 53,32 ± 18,50 |
| Fútbol | 11,74 ± 2,81 | 73,91 ± 5,37 | 14,35 ± 3,21 |
| Monólogos | 9,99 ± 1,50 | 8,83 ± 0,93 | 81,18 ± 1,62 |

Matriz de confusión 1.4.7.b

Utilizando los datos de entrenamiento solamente, el porcentaje de la clase **Fútbol** aumenta un 21,62% con respecto al test y el de la clase **Monólogos** disminuye un 12,05%, lo cuál puede resultar algo sospechoso ya que lo normal es que los resultados obtenidos hubieran sido los mismos. No obstante sigue coincidiendo en que los trozos de la clase **Noticias** se clasifican principalmente como **Monólogos**.

| Audios/Entrenamiento Identidad | Cuentos | Fútbol | Monólogos |
|-----------------------------------|---------------|---------------|----------------|
| Cuentos | 94 ± 7 | 6 ± 7 | 0 |
| Noticias | 0 | 44 ± 41 | 56 ± 41 |
| Fútbol | 3 ± 5 | 89 ± 7 | 8 ± 8 |
| Monólogos | 3 ± 5 | 0 | 97 ± 5 |

Matriz de confusión 1.4.8.b

En la Matriz de confusión 1.4.8.b puede observarse como la clasificación entre **Fútbol** y **Monólogos** de los audios de la clase **Noticias** está más igualada que en la clasificación utilizando los datos de test mostrada en la Matriz de confusión 1.4.8.a, ya que el porcentaje de **Fútbol** ha aumentado y el de **Monólogos** ha disminuido.

DECISOR DIAGONAL

| Trozos/Entrenamiento Diagonal | Cuentos | Fútbol | Monólogos |
|----------------------------------|---------------------|----------------------|---------------------|
| Cuentos | 94,55 ± 2,25 | 1,23 ± 0,93 | 4,22 ± 1,73 |
| Noticias | 26,80 ± 11,08 | 56,41 ± 17,08 | 16,79 ± 10,36 |
| Fútbol | 3,74 ± 1,76 | 88,37 ± 3,49 | 7,89 ± 2,94 |
| Monólogos | 0,99 ± 0,40 | 9,36 ± 1,00 | 89,65 ± 0,95 |

Matriz de confusión 1.4.9.b

Utilizando solo los datos de entrenamiento vemos que la distribución de la clase **Noticias** entre el resto de clases es similar a los datos de test, algo inferior en la clase **Fútbol** (56,41%) y superior en las clases **Cuentos** (26,8%) y **Monólogos** (16,79%). En el resto de clases el porcentaje es mayor.

| Audios/Entrenamiento Diagonal | Cuentos | Fútbol | Monólogos |
|----------------------------------|---------------|----------------|---------------|
| Cuentos | 99 ± 3 | 0 | 1 ± 3 |
| Noticias | 13 ± 24 | 77 ± 30 | 10 ± 12 |
| Fútbol | 5 ± 7 | 69 ± 39 | 26 ± 33 |
| Monólogos | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 1.4.10.b

Por audios la clasificación es similar, el porcentaje mayor se observa en la clase **Fútbol** (pasa de 69% a 77% de los audios) y algo inferior en **Cuentos** y **Monólogos** (13% y 10% respectivamente).

DECISOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Multidimensional | Cuentos | Fútbol | Monólogos |
|---------------------------------------|---------------------|----------------------|---------------------|
| Cuentos | 92,87 ± 2,02 | 2,22 ± 1,06 | 4,91 ± 1,69 |
| Noticias | 19,14 ± 10,92 | 62,15 ± 19,04 | 18,71 ± 11,65 |
| Fútbol | 3,95 ± 1,38 | 87,93 ± 4,06 | 8,12 ± 3,30 |
| Monólogos | 1,73 ± 0,45 | 8,76 ± 0,85 | 89,51 ± 0,99 |

Matriz de confusión 1.4.11.b

Utilizando solo los datos de entrenamiento los resultados obtenidos son los mismos que con los datos de test.

| Audios/Entrenamiento Multidimensional | Cuentos | Fútbol | Monólogos |
|---------------------------------------|------------|----------------|---------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 11 ± 17 | 75 ± 25 | 17 ± 13 |
| Fútbol | 0 | 99 ± 3 | 1 ± 3 |
| Monólogos | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 1.4.12.b

Haciendo uso únicamente de los datos de entrenamiento, los resultados son similares, principalmente los audios de la clase **Noticias** se clasifican como **Fútbol** (el porcentaje es menor que el obtenido por trozos, de 81% a 75%), seguido de **Monólogos** (11% a 17%) y de **Cuentos** (8% a 11%).

Sin Fútbol

DECISOR IDENTIDAD

| Trozos/Entrenamiento Identidad | Cuentos | Noticias | Monólogos |
|--------------------------------|---------------------|---------------------|---------------------|
| Cuentos | 91,38 ± 2,82 | 2,33 ± 0,55 | 6,29 ± 2,53 |
| Noticias | 1,33 ± 0,60 | 92,74 ± 1,61 | 5,93 ± 1,46 |
| Fútbol | 31,24 ± 5,25 | 13,65 ± 1,58 | 55,11 ± 5,30 |
| Monólogos | 10,82 ± 1,60 | 5,97 ± 1,66 | 83,21 ± 2,89 |

Matriz de confusión 1.4.13.b

Utilizando solo los datos de entrenamiento se puede ver como los resultados son bastante similares en la clase **Fútbol** y algo mejores en el resto de clases.

| Audios/Entrenamiento Identidad | Cuentos | Noticias | Monólogos |
|--------------------------------|------------|------------|----------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 10 ± 8 | 20 ± 12 | 70 ± 19 |
| Monólogos | 5 ± 7 | 1 ± 3 | 94 ± 9 |

Matriz de confusión 1.4.14.b

El porcentaje de la clase **Monólogos** aumenta con respecto a los datos observados en la Matriz de confusión 1.4.14.a, disminuyendo levemente el de **Cuentos** y **Noticias**.

DECISOR DIAGONAL

| Trozos/Entrenamiento Diagonal | Cuentos | Noticias | Monólogos |
|-------------------------------|---------------------|---------------------|----------------------|
| Cuentos | 93,86 ± 2,28 | 1,67 ± 0,64 | 4,47 ± 1,99 |
| Noticias | 0,85 ± 0,39 | 98,66 ± 0,31 | 0,49 ± 0,35 |
| Fútbol | 27,53 ± 8,98 | 16,36 ± 3,66 | 56,11 ± 10,37 |
| Monólogos | 1,99 ± 0,58 | 5,09 ± 1,96 | 92,92 ± 1,94 |

Matriz de confusión 1.4.15.b

Utilizando solo los datos de entrenamiento los resultados son los mismos en la clasificación de la clase **Fútbol** y mejores en el resto de clases.

| Audios/Entrenamiento Diagonal | Cuentos | Noticias | Monólogos |
|-------------------------------|---------------|----------------|------------|
| Cuentos | 99 ± 3 | 0 | 1 ± 3 |
| Noticias | 13 | 84 ± 25 | 3 ± 7 |
| Fútbol | 6 ± 11 | 58 ± 37 | 36 ± 34 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.16.b

A diferencia de la Matriz de confusión 1.4.16.a los audios pasan de estar clasificados como **Monólogos** a ser clasificados como **Noticias** en más de la mitad de los casos.

DECISOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Multidimensional | Cuentos | Noticias | Monólogos |
|---------------------------------------|---------------------|---------------------|----------------------|
| Cuentos | 93,57 ± 1,96 | 1,55 ± 0,60 | 4,88 ± 1,77 |
| Noticias | 0,98 ± 0,41 | 98,35 ± 0,32 | 0,67 ± 0,35 |
| Fútbol | 31,35 ± 9,41 | 16,11 ± 3,66 | 52,54 ± 10,59 |
| Monólogos | 2,60 ± 0,67 | 5,11 ± 1,75 | 92,29 ± 1,98 |

Matriz de confusión 1.4.17.b

Utilizando solo los datos de entrenamiento se obtienen los mismos resultados en la fila de la clase **Fútbol** que los obtenidos en la Matriz de confusión 1.4.17.a. Las otras tres clases consiguen mejores resultados.

| Audios/Entrenamiento Multidimensional | Cuentos | Noticias | Monólogos |
|---------------------------------------|----------------|----------------|----------------|
| Cuentos | 99 ± 3 | 0 | 1 ± 3 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 18 ± 10 | 26 ± 12 | 56 ± 17 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 1.4.18.b

En este caso los datos son similares, el porcentaje disminuye ligeramente en las clases **Cuentos** y **Monólogos**, y aumenta en la clase **Noticias**.

Sin Monólogos

DECISOR IDENTIDAD

| Trozos/Entrenamiento Identidad | Cuentos | Noticias | Fútbol |
|--------------------------------|---------------------|---------------------|---------------------|
| Cuentos | 88,72 ± 1,98 | 2,20 ± 0,64 | 9,08 ± 1,66 |
| Noticias | 1,31 ± 0,59 | 95,06 ± 0,68 | 3,63 ± 0,77 |
| Fútbol | 14,24 ± 3,60 | 4,66 ± 0,95 | 81,10 ± 3,48 |
| Monólogos | 28,66 ± 6,78 | 14,79 ± 2,94 | 56,55 ± 4,48 |

Matriz de confusión 1.4.19.b

En la fila de la clase **Monólogos** se obtiene el mismo resultado que en la Matriz de confusión 1.4.19.a. El resto de clases se consiguen porcentajes más elevados.

| Audios/Entrenamiento Identidad | Cuentos | Noticias | Fútbol |
|--------------------------------|----------------|--------------|----------------|
| Cuentos | 93 ± 8 | 0 | 7 ± 8 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 7 ± 9 | 5 ± 7 | 88 ± 15 |
| Monólogos | 23 ± 18 | 3 ± 5 | 74 ± 16 |

Matriz de confusión 1.4.20.b

Observando la clasificación por audios haciendo uso únicamente de los datos de entrenamiento, se puede ver como los resultados son bastante parecidos a los vistos en la matriz 1.4.20.a.

DECISOR DIAGONAL

| Trozos/Entrenamiento Diagonal | Cuentos | Noticias | Fútbol |
|-------------------------------|----------------------|---------------------|----------------------|
| Cuentos | 96,96 ± 1,15 | 1,64 ± 0,68 | 1,40 ± 1,15 |
| Noticias | 1 ± 0,32 | 98,58 ± 0,26 | 0,42 ± 0,22 |
| Fútbol | 4,03 ± 1,93 | 0,59 ± 0,52 | 95,38 ± 1,93 |
| Monólogos | 36,68 ± 14,69 | 6,75 ± 1,17 | 56,57 ± 13,95 |

Matriz de confusión 1.4.21.b

Utilizando solo los datos de entrenamiento se obtienen los mismos resultados que usando los datos de test.

| Audios/Entrenamiento Diagonal | Cuentos | Noticias | Fútbol |
|-------------------------------|----------------|------------|----------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 0 | 0 | 100 |
| Monólogos | 33 ± 25 | 0 | 67 ± 25 |

Matriz de confusión 1.4.22.b

De nuevo se consiguen los mismos resultados utilizando solamente los datos de entrenamiento. El resto de clases consigue clasificar correctamente la totalidad de los audios.

DECISOR MULTIDIMENSIONAL

| Trozos/Entrenamiento Multidimensional | Cuentos | Noticias | Fútbol |
|---------------------------------------|----------------------|---------------------|---------------------|
| Cuentos | 96,27 ± 1,20 | 1,44 ± 0,51 | 2,29 ± 1,05 |
| Noticias | 1,02 ± 0,34 | 98,29 ± 0,24 | 0,69 ± 0,27 |
| Fútbol | 4,88 ± 2,02 | 0,54 ± 0,53 | 94,58 ± 2,19 |
| Monólogos | 42,87 ± 13,34 | 7,13 ± 1,20 | 50 ± 12,66 |

Matriz de confusión 1.4.23.b

Los resultados son los mismos que los obtenidos en la Matriz de confusión 1.4.23.a.

| Audios/Entrenamiento Multidimensional | Cuentos | Noticias | Fútbol |
|---------------------------------------|----------------|------------|----------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 0 | 0 | 100 |
| Monólogos | 42 ± 22 | 0 | 58 ± 22 |

Matriz de confusión 1.4.24.b

Utilizando únicamente los datos de entrenamiento, los **Monólogos** son clasificados mayoritariamente como **Fútbol** (58%) seguido de **Cuentos** (42%), al contrario que sucedía en la Matriz de confusión 1.4.24.a, donde el mayor porcentaje lo recibía la clase **Cuentos** (52%).

CLASIFICADOR KNN

Experimento 1

| Trozos/Entrenamiento | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|----------------------|
| Cuentos | 94,64 ± 1,38 | 0,34 ± 0,25 | 1,22 ± 0,88 | 3,8 ± 1,01 |
| Noticias | 1,96 ± 0,95 | 90,25 ± 1,35 | 2,76 ± 1,24 | 5,03 ± 0,80 |
| Fútbol | 1,14 ± 0,32 | 0,26 ± 0,38 | 93,05 ± 1,37 | 5,55 ± 1,13 |
| Monólogos | 1,43 ± 0,65 | 0,54 ± 0,22 | 3,01 ± 0,92 | 95,02 ± 1,33, |

Matriz de confusión 2.1.1.b

Utilizando únicamente los datos de entrenamiento el porcentaje de acierto es bastante superior al obtenido utilizando también los datos de test en las cuatro clases, por lo que se podría sospechar que se está sobreajustando.

Las desviaciones típicas son muy pequeñas, todos los valores se encuentran muy próximos a la media.

| Audios/Entrenamiento | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------|------------|------------|------------|------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 2.1.2.b

Utilizando únicamente los datos de entrenamiento la totalidad de los audios es clasificada de forma correcta. La única diferencia con respecto a la clasificación por trozos es que esta vez no es tan llamativa la diferencia de aciertos, ya que por audios se obtenían mejores resultados a la hora de clasificar.

Experimento 2

| Trozos/Entrenamiento | Cuentos | Noticias | Fútbol | Monólogos |
|----------------------|--------------|--------------|--------------|--------------|
| Cuentos | 90,73 | 0,56 | 2,53 | 6,18 |
| Noticias | 2,22 | 89,33 | 1,56 | 6,89 |
| Fútbol | 2,04 | 0,25 | 91,33 | 6,38 |
| Monólogos | 2,48 | 0,23 | 2,48 | 94,81 |

Matriz de confusión 2.2.1.b

Haciendo uso solamente de los datos de entrenamiento mejoran considerablemente los porcentajes de acierto, pero dentro de los límites lógicos, por lo que se puede afirmar que el clasificador no está sobreajustando. Comparando con los resultados del experimento anterior se puede ver como, al contrario que pasaba con los datos de test, en los que se obtenían mejores resultados, con los datos de entrenamiento los porcentajes de acierto son inferiores.

Experimento 3

| Trozos/Entrenamiento Aleatorio | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 80,71 ± 1,84 | 0,64 ± 0,30 | 5,56 ± 1,17 | 13,09 ± 1,61 |
| Noticias | 3,47 ± 0,73 | 79,41 ± 1,77 | 5,91 ± 0,94 | 11,21 ± 1,64 |
| Fútbol | 4,46 ± 1,29 | 0,77 ± 0,60 | 78,32 ± 2,51 | 16,45 ± 1,72 |
| Monólogos | 4,4 ± 1,23 | 1,45 ± 0,67 | 8,33 ± 1,22 | 85,82 ± 2,32 |

Matriz de confusión 2.3.1.b

Haciendo uso únicamente de los datos de entrenamiento los resultados no mejoran los calculados utilizando los datos de test. La única clase que obtiene un porcentaje ligeramente superior es **Noticias** (un 0,62% más).

Experimento 4

Sin Cuentos

| Trozos/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 3,76 ± 1,19 | 29,53 ± 7,99 | 66,71 ± 8,70 |
| Noticias | 91,63 ± 1,36 | 2,83 ± 1,36 | 5,54 ± 0,78 |
| Fútbol | 0,29 ± 0,45 | 93,58 ± 1,49 | 6,13 ± 1,25 |
| Monólogos | 0,65 ± 0,31 | 3,03 ± 0,90 | 96,32 ± 1,16 |

Matriz de confusión 2.4.1.b

Las soluciones utilizando solo los datos de entrenamiento no varían, los trozos se reparten de la misma manera que utilizando la mitad de los datos para entrenar y la mitad para test.

| Audios/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|----------------|----------------|----------------|
| Cuentos | 0 | 14 ± 14 | 86 ± 14 |
| Noticias | 81 ± 29 | 9 ± 29 | 10 ± 7 |
| Fútbol | 0 | 65 ± 18 | 35 ± 18 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.2.b

Utilizando solo los datos de entrenamiento aumenta el porcentaje de las clases **Fútbol** y **Monólogos** y disminuye en de la clase **Cuentos**.

Sin Noticias

| Trozos/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 94,71 ± 1,44 | 1,32 ± 1,06 | 3,97 ± 0,89 |
| Noticias | 16,77 ± 3,54 | 23,98 ± 3,70 | 59,25 ± 3,98 |
| Fútbol | 1,16 ± 0,33 | 93,19 ± 1,16 | 5,65 ± 1,12 |
| Monólogos | 1,46 ± 0,71 | 2,96 ± 0,85 | 95,58 ± 1,12 |

Matriz de confusión 2.4.3.b

La clase Noticias se clasifica principalmente como **Monólogos**, al igual que se vio en la Matriz de Confusión 2.4.3.a, aunque en un porcentaje algo mayor (de 57,33% a 59,25%). Los porcentajes de acierto en el resto de clases aumentan considerablemente.

| Audios/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|---------------|----------------|---------------|
| Cuentos | 90 ± 8 | 0 | 10 ± 8 |
| Noticias | 1 ± 3 | 12 ± 6 | 87 ± 5 |
| Fútbol | 9 ± 3 | 67 ± 13 | 24 ± 11 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.4.b

En el caso de la clasificación por audios, el porcentaje de clasificación como **Monólogos** aumenta considerablemente al conseguido introduciendo los datos de test.

Sin Fútbol

| Trozos/Entrenamiento | Cuentos | Noticias | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 95,61 ± 0,88 | 0,32 ± 0,27 | 4,07 ± 0,94 |
| Noticias | 2,15 ± 1,02 | 91,99 ± 1,03 | 5,86 ± 0,87 |
| Fútbol | 23,8 ± 3,51 | 4,59 ± 1,46 | 71,61 ± 3,66 |
| Monólogos | 1,70 ± 0,78 | 0,70 ± 0,46 | 97,60 ± 1,00 |

Matriz de confusión 2.4.5.b

Los resultados son similares a los obtenidos en la Matriz de confusión 2.4.5.a, disminuyendo ligeramente el porcentaje de trozos de la clase **Fútbol** que se clasifican como **Monólogos** y **Cuentos** y aumentando el de **Noticias**. Los porcentajes del resto de clases aumentan ampliamente.

| Audios/Entrenamiento | Cuentos | Noticias | Monólogos |
|----------------------|---------------|---------------|------------|
| Cuentos | 89 ± 9 | 3 ± 5 | 8 ± 9 |
| Noticias | 0 | 77 ± 5 | 23 ± 5 |
| Fútbol | 0 | 0 | 100 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 2.4.6.b

Por audios, al igual que introduciendo los datos de test, con solo los datos de entrenamiento, los 10 audios de la clase **Fútbol** se clasifican como **Monólogos**.

Sin Monólogos

| Trozos/Entrenamiento | Cuentos | Noticias | Fútbol |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 97,94 ± 0,97 | 0,37 ± 0,32 | 1,69 ± 1,09 |
| Noticias | 2,18 ± 0,88 | 94,63 ± 1,36 | 3,19 ± 1,03 |
| Fútbol | 1,85 ± 0,58 | 0,44 ± 0,37 | 97,71 ± 0,85 |
| Monólogos | 33,82 ± 6,92 | 11,37 ± 2,57 | 54,81 ± 5,29 |

Matriz de confusión 2.4.7.b

Teniendo en cuenta solo los datos de entrenamiento se puede comprobar que la clasificación es la misma con respecto a la clase **Monólogos**, se clasifican principalmente como **Fútbol**, aunque en un porcentaje levemente inferior (de 55,75% a 54,81%).

| Audios/Test | Cuentos | Noticias | Fútbol |
|-------------|---------------|------------|---------------|
| Cuentos | 99 ± 3 | 1 ± 3 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 10 | 1 ± 3 | 89 ± 3 |
| Monólogos | 13 ± 6 | 0 | 87 ± 6 |

Matriz de confusión 2.4.8.a

Usando solo los datos de entrenamiento, el porcentaje de la clase **Fútbol** desciende, aumentando ligeramente el de la clase **Cuentos**.

CLASIFICADOR SVM

Experimento 1

KERNEL LINEAL

| Trozos/Entrenamiento Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 94,99 ± 1,49 | 0,14 ± 0,21 | 2,76 ± 1,33 | 2,11 ± 1,23 |
| Noticias | 0,27 ± 0,33 | 98,40 ± 0,61 | 0,49 ± 0,29 | 0,84 ± 0,59 |
| Fútbol | 3,88 ± 1,17 | 0,76 ± 0,47 | 90,89 ± 2,45 | 4,47 ± 1,46 |
| Monólogos | 3,18 ± 0,80 | 0,90 ± 0,61 | 2,89 ± 0,83 | 93,03 ± 1,51 |

Matriz de confusión 3.1.1.b

Utilizando únicamente los datos de entrenamiento se ve como mejoran, como era de esperar, todos los porcentajes, siendo la clase **Noticias** la que tiene un mejor resultado (98,4%), seguida esta vez de la clase **Cuentos** (94,99%), **Monólogos** (93,03%) y **Fútbol** (90,89%). Si bien los resultados son mejores, aún comete errores con respecto al resto de clases, por lo que no se está sobreajustando.

Se puede observar como las desviaciones se ven suavizadas al utilizar solo los datos de entrenamiento. La dispersión más elevada es la obtenida en la diagonal principal en la clase Fútbol, al igual que pasaba utilizando la mitad de los datos de test.

| Audios/Entrenamiento Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|-----------------------------|------------|------------|------------|------------|
| Cuentos | 100 | 0 | 0 | 0 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 0 | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 3.1.2.b

Haciendo uso solamente de los datos de entrenamiento en la clasificación por audios, se clasifican todos los audios de forma correcta, por esto mismo las desviaciones típicas son nulas. No sobreajusta puesto que los resultados obtenidos utilizando también los datos de test (Matriz de confusión 3.1.2.a) no eran mucho peores, ya que a excepción de la clase **Fútbol**, el resto de clases superaban el 90% de aciertos.

KERNEL GAUSSIANO

| Trozos/Entrenamiento Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 80,61 ± 7,98 | 0,82 ± 0,45 | 7,92 ± 3,54 | 10,65 ± 6,97 |
| Noticias | 1,27 ± 0,71 | 91,99 ± 0,87 | 0,82 ± 0,73 | 5,92 ± 0,67 |
| Fútbol | 7,79 ± 3,70 | 1,42 ± 0,85 | 70,08 ± 8,00 | 20,71 ± 6,89 |
| Monólogos | 4,69 ± 2,62 | 2,31 ± 0,79 | 2,71 ± 1,20 | 90,29 ± 3,89 |

Matriz de confusión 3.1.3.b

Haciendo uso únicamente de los datos de entrenamiento se comprueba como los porcentajes de acierto de las cuatro clases se han visto incrementados, siendo **Noticias** la clase mejor clasificada. La mejora más significativa la consigue la clase **Fútbol**, con un 29,17% más de aciertos que por trozos (Matriz de confusión 3.1.3.a), aunque continúa siendo la clase con el peor porcentaje. Aunque los resultados en general sean mejores, todavía existen porcentajes de error importantes. Por ejemplo, en la fila de la clase **Fútbol** se puede ver un porcentaje importante de fallos con respecto a la clase **Monólogos** (20,71%) y otro menor aunque significativo con la clase **Cuentos** (7,79%). En la clase **Cuentos** también destacan dos porcentajes de error con respecto a las clases **Fútbol** y **Monólogos** (7,92% y 10,65% respectivamente). Por todo esto se dirá que el clasificador no sobreajusta.

En general, las desviaciones típicas se han visto moderadas haciendo uso solamente de los datos de entrenamiento. La única dispersión que se ha visto incrementada podemos verla en la fila de la clase **Cuentos**, en relación al porcentaje de aciertos, ya que de 4,75% ha pasado a un 7,98%.

| Audios/Entrenamiento Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------|---------------|------------|----------------|------------|
| Cuentos | 92 ± 6 | 0 | 6 ± 5 | 2 ± 4 |
| Noticias | 0 | 100 | 0 | 0 |
| Fútbol | 2 ± 4 | 0 | 87 ± 13 | 11 ± 11 |
| Monólogos | 0 | 0 | 0 | 100 |

Matriz de confusión 3.1.4.b

En la clasificación por audios usando solo los datos de entrenamiento se ven incrementados los aciertos tanto para la clase **Cuentos** como para la clase **Fútbol** en un 21,05% y 26,08% respectivamente. Como se puede observar, las desviaciones típicas de la clase **Cuentos** se han visto reducidas considerablemente. En cambio, para la clase **Fútbol** han descendido levemente.

Experimento 2

KERNEL LINEAL

| Trozos/Entrenamiento Gaussiano/ | Cuentos | Noticias | Fútbol | Monólogos |
|---------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 93,54 | 0,28 | 3,37 | 2,81 |
| Noticias | 0,44 | 97,78 | 0,22 | 1,56 |
| Fútbol | 4,34 | 1,02 | 89,79 | 4,85 |
| Monólogos | 2,93 | 1,13 | 3,39 | 92,55 |

Matriz de confusión 3.2.1.b

Mejoran los resultados de las cuatro clases con respecto a la matriz 3.2.1.a, siendo de nuevo **Noticias** la clase mejor clasificada y **Fútbol** la peor. Se puede decir que los porcentajes de mejora con respecto a las soluciones obtenidas utilizando los datos de test son razonables, por lo que se afirmará que el clasificador no está sobreentrenado, es decir, no sobreajusta y sería válido para la clasificación que se quiere llevar a cabo.

KERNEL GAUSSIANO

| Trozos/Entrenamiento Mitad/Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|--------------------------------------|--------------|--------------|--------------|--------------|
| Cuentos | 82,02 | 0,84 | 8,43 | 8,71 |
| Noticias | 1,78 | 89,78 | 0,44 | 8 |
| Fútbol | 10,2 | 1,79 | 58,93 | 29,08 |
| Monólogos | 4,74 | 2,26 | 1,58 | 91,42 |

Matriz de confusión 3.2.2.b

Utilizando únicamente los datos de entrenamiento se observa como la clase mejor clasificada pasa a ser **Monólogos**, en vez de **Noticias** como haciendo uso de los datos de test con un 91,42% (2,61% más aciertos). La clase peor clasificada continúa siendo **Fútbol**, y llama la atención que el porcentaje sea menor que utilizando los datos de test (58,93%), el porcentaje de aciertos también ha disminuido para la clase **Noticias** un 2,17%. La clase **Cuentos** sube un 5,12% y la clase **Noticias** un 2,2%. En general, los errores disminuyen discretamente.

Experimento 3

KERNEL LINEAL

| Trozos/Entrenamiento Aleatorio/Lineal | Cuentos | Noticias | Fútbol | Monólogos |
|---------------------------------------|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 93,63 ± 0,95 | 0,14 ± 0,20 | 4,49 ± 0,88 | 1,74 ± 0,63 |
| Noticias | 0,34 ± 0,22 | 98,13 ± 0,68 | 0,42 ± 0,22 | 1,11 ± 0,59 |
| Fútbol | 5,43 ± 1,01 | 0,84 ± 0,42 | 88,17 ± 1,52 | 5,56 ± 0,85 |
| Monólogos | 3,32 ± 0,66 | 0,86 ± 0,39 | 3,79 ± 0,70 | 92,03 ± 0,85 |

Matriz de confusión 3.3.1.b

Utilizando solo los datos de entrenamiento se puede ver como los porcentajes de la diagonal principal han aumentado y todos los valores situados fuera de ella, es decir, los porcentajes de errores, han descendido.

Las desviaciones típicas en los datos de entrenamiento son más pequeñas que las que obteníamos utilizando los datos de test, por lo que los datos se concentran aún más alrededor de la media.

KERNEL GAUSSIANO

| Trozos/Entrenamiento Aleatorio/Gaussiano | Cuentos | Noticias | Fútbol | Monólogos |
|--|---------------------|---------------------|---------------------|---------------------|
| Cuentos | 80,90 ± 2,05 | 1,07 ± 0,62 | 8,96 ± 1,13 | 9,07 ± 1,84 |
| Noticias | 1,25 ± 0,60 | 91,04 ± 1,24 | 0,53 ± 0,26 | 7,18 ± 1,04 |
| Fútbol | 10 ± 1,08 | 1,86 ± 0,56 | 63,38 ± 2,92 | 24,76 ± 2,16 |
| Monólogos | 4,56 ± 1,21 | 2,71 ± 0,81 | 1,62 ± 0,44 | 91,11 ± 1,34 |

Matriz de confusión 3.3.2.b

Utilizando únicamente los datos de entrenamiento, mejoran ligeramente los resultados de las cuatro clases: **Cuentos** un 3,00%, **Noticias** un 0,77%, **Fútbol** un 0,34% y **Monólogos** un 1,91%. Que no mejore en exceso puede deberse a que los trozos que se están cogiendo del audio de manera aleatoria para testear son muestras suficientemente significativas.

Las desviaciones típicas son similares a los obtenidas utilizando los datos de test (Matriz de confusión 3.3.2.a), únicamente destacan los resultados conseguidos en la clase **Fútbol** con respecto a los aciertos, 2,92%, y a los fallos, 2,16%.

Experimento 4

Sin Cuentos

KERNEL LINEAL

| Trozos/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 8,15 ± 2,15 | 41,07 ± 9,36 | 50,78 ± 9,72 |
| Noticias | 98,49 ± 0,58 | 0,51 ± 0,33 | 1 ± 0,59 |
| Fútbol | 0,81 ± 0,55 | 94,42 ± 1,76 | 4,77 ± 1,57 |
| Monólogos | 0,98 ± 0,63 | 3,70 ± 0,86 | 95,32 ± 1,17 |

Matriz de confusión 3.4.1.b

Utilizando solo los datos de entrenamiento, los trozos de la clase **Cuentos** se reparten de la misma manera, y los porcentajes de las otras tres clases mejoran considerablemente.

| Audios/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|------------|------------|----------------|
| Cuentos | 0 | 46 ± 14 | 54 ± 14 |
| Noticias | 100 | 0 | 0 |
| Fútbol | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.2.b

En la clasificación por audios se ven suavizadas las desviaciones típicas y se ven incrementados los porcentajes con respecto a la clase **Fútbol** (de 41% a 46%) y **Monólogos** (de 49% a 54%). Sobre la clase **Noticias** no recae ningún audio. En el resto de clases se puede ver como se clasifican correctamente todos los audios.

KERNEL GAUSSIANO

| Trozos/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|----------------------|
| Cuentos | 3,78 ± 1,48 | 32,87 ± 14,82 | 63,35 ± 15,19 |
| Noticias | 92,38 ± 0,81 | 1,11 ± 0,73 | 6,51 ± 0,91 |
| Fútbol | 1,47 ± 0,93 | 75,67 ± 7,09 | 22,86 ± 7,19 |
| Monólogos | 2,31 ± 0,79 | 3,45 ± 1,46 | 94,24 ± 1,97 |

Matriz de confusión 3.4.3.b

Haciendo uso únicamente de los datos de entrenamiento, los porcentajes de las tres clases obtienen mejores resultados.

| Audios/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|-------------|-------------|-------------|
| Cuentos | 4 ± 13 | 34 ± 19 | 62 ± 15 |
| Noticias | 90 ± 32 | 10 ± 32 | 0 |
| Fútbol | 0 | 80 ± 30 | 20 ± 30 |
| Monólogos | 0 | 1 ± 3 | 99 ± 3 |

Matriz de confusión 3.4.4.b

Utilizando solamente los datos de entrenamiento sube el porcentaje de **Cuentos** clasificados como **Monólogos** y baja el porcentaje de la clase **Fútbol** y **Noticias**. Como se puede observar, al igual que sucedió en el Kernel Lineal, los valores de las desviaciones típicas son elevados y confirman que los **Cuentos** son clasificados como **Monólogos** y **Fútbol** primordialmente.

Sin Noticias

KERNEL LINEAL

| Trozos/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|------------------|-------------------|------------------|
| Cuentos | $94,94 \pm 1,46$ | $2,76 \pm 1,33$ | $2,30 \pm 1,20$ |
| Noticias | $14,51 \pm 3,94$ | $15,69 \pm 10,75$ | $69,80 \pm 8,95$ |
| Fútbol | $3,88 \pm 1,17$ | $91,1 \pm 2,44$ | $5,02 \pm 1,51$ |
| Monólogos | $3,22 \pm 0,77$ | $2,94 \pm 0,89$ | $93,84 \pm 1,05$ |

Matriz de confusión 3.4.5.b

Con respecto a la clasificación de la clase **Noticias** utilizando únicamente los datos de entrenamiento apenas varían los datos, aumentan ligeramente los porcentajes de **Cuentos** y **Fútbol** y disminuye, también ligeramente, el de la clase **Monólogos**.

| Audios/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|------------|-------------|-------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 1 ± 2 | 17 ± 33 | 85 ± 32 |
| Fútbol | 0 | 100 | 0 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.6.b

Utilizando solo los datos de entrenamiento aumenta el porcentaje de la clase **Cuentos** y disminuye el de la clase **Monólogos**. Los valores de las desviaciones típicas también aumentan considerablemente.

KERNEL GAUSSIANO

| Trozos/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 80,84 ± 8,17 | 7,95 ± 3,50 | 11,21 ± 7,28 |
| Noticias | 6,35 ± 3,94 | 7,58 ± 6,67 | 86,07 ± 5,48 |
| Fútbol | 7,79 ± 3,70 | 70,65 ± 8,35 | 21,56 ± 7,10 |
| Monólogos | 4,71 ± 2,60 | 2,83 ± 1,29 | 92,46 ± 3,44 |

Matriz de confusión 3.4.7.b

En la matriz de confusión de los datos de entrenamiento vemos como los resultados son los mismos con respecto a la clase **Noticias**. Los resultados con respecto a las otras tres clases son mejores.

| Audios/Entrenamiento | Cuentos | Fútbol | Monólogos |
|----------------------|---------------|----------------|---------------|
| Cuentos | 90 ± 7 | 8 ± 6 | 2 ± 4 |
| Noticias | 0 | 1 ± 3 | 99 ± 3 |
| Fútbol | 2 ± 4 | 87 ± 13 | 11 ± 11 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.8.b

Observando la fila de la clase Noticias se puede ver como haciendo uso solo de los datos de entrenamiento estas son clasificadas como **Monólogos** en prácticamente su totalidad con excepción de un pequeño porcentaje (1%) que se clasifica como **Fútbol**. Las desviaciones típicas se han visto reducidas con respecto a las de la Matriz de confusión 3.4.8.a. Las otras tres clases obtienen porcentajes más elevados.

Sin Fútbol

KERNEL LINEAL

| Trozos/Entrenamiento | Noticias | Fútbol | Monólogos |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 96,84 ± 1,31 | 0,14 ± 0,21 | 3,02 ± 1,33 |
| Noticias | 0,36 ± 0,32 | 98,71 ± 0,64 | 0,93 ± 0,56 |
| Fútbol | 32,45 ± 5,42 | 7,93 ± 2,55 | 59,62 ± 7,10 |
| Monólogos | 3,29 ± 0,99 | 0,88 ± 0,61 | 95,83 ± 1,46 |

Matriz de confusión 3.4.9.b

Solo con los datos de entrenamiento se obtienen exactamente los mismos resultados en la clase **Fútbol**. El resto de clases obtiene porcentajes más elevados que los obtenidos en la Matriz de confusión 3.4.9.a.

| Audios/Entrenamiento | Cuentos | Noticias | Monólogos |
|----------------------|--------------|--------------|---------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 8 ± 8 | 1 ± 3 | 91 ± 7 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.10.b

Haciendo uso solamente de los datos de entrenamiento, los audios de la clase Fútbol clasificados como Cuentos se ven reducidos (de 19% a 8%) en aumento de la clase Monólogos (de 81% a 91%). El resto de clases consigue clasificar la totalidad de los audios de forma correcta.

KERNEL GAUSSIANO

| Trozos/Entrenamiento | Cuentos | Noticias | Monólogos |
|----------------------|----------------------|----------------------|----------------------|
| Cuentos | 62,26 ± 40,77 | 11,21 ± 19,15 | 26,53 ± 25,80 |
| Noticias | 28,67 ± 44,03 | 64,69 ± 44,24 | 6,64 ± 0,76 |
| Fútbol | 16,56 ± 10,97 | 25,26 ± 34,22 | 58,18 ± 24,11 |
| Monólogos | 4,53 ± 2,86 | 2,85 ± 1,21 | 92,62 ± 3,04 |

Matriz de confusión 3.4.11.b

Utilizando los datos de entrenamiento, el porcentaje más alto sigue siendo para **Monólogos**, aunque este se vea reducido comparado con el mostrado en la Matriz de confusión 3.4.11.b (58,18% frente al 66,42% que se conseguía con los datos de test). Desciende también el porcentaje de la clase **Cuentos**, aumentando el porcentaje con respecto a la clase **Noticias**.

| Audios/Entrenamiento | Cuentos | Noticias | Monólogos |
|----------------------|---------------|------------|---------------|
| Cuentos | 98 ± 4 | 0 | 2 ± 4 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 3 ± 5 | 0 | 97 ± 5 |
| Monólogos | 0 | 0 | 100 |

Matriz de confusión 3.4.12.b

Por audios aumenta el porcentaje con respecto a la clase **Monólogos** disminuyendo el de la clase **Cuentos**.

Sin Monólogos

KERNEL LINEAL

| Trozos/Entrenamiento | Cuentos | Noticias | Fútbol |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 96,41 ± 1,43 | 0,29 ± 0,24 | 3,30 ± 1,36 |
| Noticias | 0,13 ± 0,15 | 99,29 ± 0,27 | 0,58 ± 0,30 |
| Fútbol | 4,23 ± 1,51 | 0,52 ± 0,41 | 95,25 ± 1,53 |
| Monólogos | 30,81 ± 6,16 | 19,72 ± 1,76 | 49,47 ± 6,14 |

Matriz de confusión 3.4.13.b

Utilizando solo los datos de entrenamiento, se obtienen los mismos resultados con respecto a la clase **Monólogos**. El resto de clases obtienen porcentajes más elevados.

| Audios/Entrenamiento | Cuentos | Noticias | Fútbol |
|----------------------|----------------|----------------|----------------|
| Cuentos | 100 | 0 | 0 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 0 | 0 | 100 |
| Monólogos | 24 ± 16 | 12 ± 10 | 64 ± 24 |

Matriz de confusión 3.4.14.b

Usando solamente los datos de entrenamiento, disminuye el porcentaje con respecto a la clase **Fútbol** y aumenta el de la clase **Cuentos**.

KERNEL GAUSSIANO

| Trozos/Entrenamiento | Cuentos | Noticias | Fútbol |
|----------------------|---------------------|---------------------|---------------------|
| Cuentos | 87,48 ± 4,81 | 1,37 ± 0,61 | 11,15 ± 4,44 |
| Noticias | 1,71 ± 0,82 | 96,24 ± 0,91 | 2,05 ± 1,15 |
| Fútbol | 9,19 ± 4,13 | 2,37 ± 0,98 | 88,44 ± 4,57 |
| Monólogos | 21,31 ± 5,86 | 19,90 ± 3,14 | 58,79 ± 5,36 |

Matriz de confusión 3.4.15.b

Los resultados de la clase Monólogos utilizando solo los datos de entrenamiento, son los mismos que con los datos de test. Con el resto de clases si varían, aumentando los porcentajes en todas ellas.

| Audios/Entrenamiento | Cuentos | Noticias | Fútbol |
|----------------------|---------------|------------|---------------|
| Cuentos | 89 ± 7 | 0 | 11 ± 7 |
| Noticias | 0 | 100 | 0 |
| Fútbol | 2 ± 4 | 0 | 98 ± 4 |
| Monólogos | 15 ± 15 | 11 ± 6 | 74 ± 17 |

Matriz de confusión 3.4.16.b

Utilizando solo los datos de entrenamiento, disminuye el porcentaje de la clase **Fútbol**, y aumentan los de las clases **Cuentos** y **Noticias**.

BIBLIOGRAFÍA

- [1] L. R. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. 2003.
- [2] Anders Meng, Peter Ahrendt, Jan Larsen, Senior Member, IEEE, and Lars Kai Hansen. *Temporal Feature Integration for Music Genre Classification*, July 2007.
- [3] Beth Logan. *Mel Frequency cepstral coefficients for music modelling*. Cambridge Research Laboratory, 2000.
- [4] Sigurdur Sigurdsson, Kaare Brandt Petersen, and Tue Lehn-Schioler. *Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music*. Pro- Intl. Symp. Music Information Retrieval (ISMIR), 2006.
- [5] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*. 2001. Software en <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [6] Vladimir Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.
- [7] Tom M. Mitchell. *Machine Learning*. McGraw-Hill International Edition. 1997.
- [8] Igor Kononenko. *Introduction to Machine Learning*. The MIT Press, 2004.
- [9] Tobias Glasmachers, Christian Igel, Senior Member, IEEE. *Maximum Likelihood Model Selection for 1-Norm Soft Margin SVMs with Multiple Parameters*, April 2010.
- [10] Jordi Porta Zamorano. *Clasificación de patrones: Métodos supervisados*. Escuela Politécnica Superior. Universidad Autónoma de Madrid, April 2005.
- [11] Fernando Fernández Rebollo. *Teoría de Decisión Bayesiana*. 2002.
- [12] *Manual de uso de Matlab*. Universidad de Oviedo. EPI de Gijón. 2010.
- [13] *Manual básico de Matlab*. Apoyo a investigación C. P. D. Servicios informáticos U.C.M. 2010.

- [14] Corinna Cortés, Vladimir Vapnik. *Support- Vector Networks*. Machine Learning, 20, 273, 297 (1995)
- [15] Christopher J.C. Burges. *A tutorial on Support Vector Machines for Pattern Recognition*. 1998.
- [16] Angulo Bahón. *Clasificación Multiclase con Máquinas de Soporte Vectorial*. www.tdx.cat/bitstream/10803/6178/.../05Capitulo4.pdf. 2001.
- [17] Gustavo A. Betancourt. *Las máquinas de Soporte Vectorial (SVMs)*. Universidad Tecnológica de Pereira. 2005.
- [18] Jesús Cid Sueiro, Vanessa Gómez Verdejo. *Apuntes de decisión. Teoría Moderna de la Detección y Estimación*. 2013.
- [19] Filippo Chieco, Carlos Pérez Pérez, Joana Rodríguez Luque. *Algoritmo del vecino más cercano aplicado a la web Filmaffinity.com*. Universidad Carlos III de Madrid. 2010.
- [20] *Extracción de características*.
<http://bibing.us.es/proyectos/abreprov/12054/fichero/MEMORIA%252F8.Cap%EDtulo+3.pdf>
- [21] Sigurdur Sigurdsson, Kaare Brandt Petersen and Tue Lehn-Schiøler. *Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music*. Technical University of Denmark. 2006.
- [22] Vladimir N. Vapnik. *The nature of Statistical Learning Theory*. New York: Springer, 1995.
- [23] Fernando Valle Padilla. *Implementación eficiente de clasificadores prior-SVM para MATLAB*. Proyecto final de carrera. Universidad Carlos III de Madrid. Julio 2011.
- [24] Rubén Solera Ureña, Fernando Pérez Cruz, Fernando Díaz de María. *Estimación de probabilidades a posteriori en SVMs multiclase para reconocimiento de habla continua*. Universidad Carlos III de Madrid. 2006.
- [25] Darío García García, Emilio Parrado Hernández, Jerónimo Arenas García y Fernando Díaz de María. *Music genre classification using temporal structure songs*. MLSP, 2010.
- [26] John Shawe Taylor, Nello Cristianini. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

- [27] J. Am. J. Shao. *Linear model selection by cross-validation*. 1993
- [28] Louis A. Liporace. *Maximum likelihood estimation for multivariate observations of markov sources*. IEEE Information Theory Society. Septiembre 1982
- [29] Andrew R. Webb, Keith D. Copsey. *Statistical Pattern Recognition*, 2011
- [30] Sigurdur Sigurdsson, Kaare Brandt Petersen, Tue Lehn Schioler. *Mel Frequency cepstral coefficients: An evaluation of robustness of MP3 encoded music*. Music Information Retrieval (ISMIR). 2006.
- [31] Cory McKay. *Issues in Automatic Musical Genre Classification*. Faculty of Music. McGill University. 2004.
- [32] Yoko Anan, Kohei Hatano, Hideo Bannai, Masayuki Takeda. *Music genre classification using similarity functions*. Department of Informatics, Kyushu University. 2011.
- [33] George Tzanetakis. Georg Essl. Perry Cook. *Automatic Musical Genre Classification of Audio Signals*. Computer Science Department. 2002.
- [34] George Tzanetakis. Perry Cook. *Musical Genre Classification of Audio Signals*. IEEE Transactions on speech and audio processing. Julio 2002.

